

(19) 世界知的所有権機関  
国際事務局



(43) 国際公開日  
2003 年 8 月 14 日 (14.08.2003)

PCT

(10) 国際公開番号  
WO 03/067527 A1

(51) 国際特許分類<sup>7</sup>: G06T 17/00, 17/40, G06F 17/50

(21) 国際出願番号: PCT/JP02/01000

(22) 国際出願日: 2002 年 2 月 6 日 (06.02.2002)

(25) 国際出願の言語: 日本語

(26) 国際公開の言語: 日本語

(71) 出願人 (米国を除く全ての指定国について): デジタルプロセス株式会社 (DIGITAL PROCESS LTD.) [JP/JP]; 〒243-0018 神奈川県 厚木市中町 2 丁目 9 番 6 号 Kanagawa (JP).

(72) 発明者; および

(75) 発明者/出願人 (米国についてのみ): 森 博己

(MORI,Hiromi) [JP/JP]; 〒243-0018 神奈川県 厚木市中町 2 丁目 9 番 6 号 デジタルプロセス株式会社内 Kanagawa (JP).

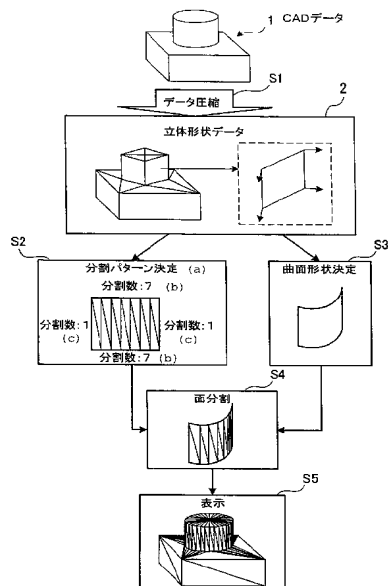
(74) 代理人: 服部 毅蔵 (HATTORI,Kiyoshi); 〒192-0082 東京都 八王子市東町 9 番 8 号 G E エジソンビル八王子 服部特許事務所 Tokyo (JP).

(81) 指定国 (国内): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.

[続葉有]

(54) Title: THREE-DIMENSIONAL SHAPE DISPLAYING PROGRAM, THREE-DIMENSIONAL SHAPE DISPLAYING METHOD, AND THREE-DIMENSIONAL SHAPE DISPLAYING DEVICE

(54) 発明の名称: 立体形状表示プログラム、立体形状表示方法、および立体形状表示装置



1...CAD DATA  
S1...DATA COMPRESSION  
2...THREE-DIMENSIONAL SHAPE DATA  
(a)...DETERMINATION OF DIVISION PATTERN  
(b)...NUMBER OF DIVISIONS : 7  
(c)...NUMBER OF DIVISIONS : 1  
S2...DETERMINATION OF DIVISION PATTERN  
S3...DETERMINATION OF CURVED SURFACE SHAPE  
S4...SURFACE DIVISION  
S5...DISPLAY

(57) Abstract: A three-dimensional shape is rapidly displayed based on compressed data. A division pattern of surfaces constituting a three-dimensional shape is determined based on three-dimensional shape data (2) defining the three-dimensional shape by at least one surface (Step S2). At the same time, the curved surface shape is determined based on the position of an apex of the surface which is defined in advance by the three-dimensional shape data (2) and the direction of the normal at the apex (Step S3). Next, the determined curved surface shape is divided according to the determined division pattern to generate surfaces along the curved surface shape (Step S4). The three-dimensional shape constituted by the generated surfaces is displayed (Step S5).

[続葉有]



(84) 指定国 (広域): ARIPO 特許 (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア特許 (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI 特許 (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

添付公開書類:  
— 国際調査報告書

2 文字コード及び他の略語については、定期発行される各 *PCT* ガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

(57) 要約:

圧縮されたデータに基づいて高速に立体形状を表示することができるようにする。少なくとも 1 つの面で立体形状を定義した立体形状データ (2) に基づいて、立体形状を構成する面の分割パターンを決定する (ステップ S 2)。同時に、立体形状データ (2) に予め定義された面の頂点の位置とその頂点における法線方向とに基づいて面の曲面形状を決定する (ステップ S 3)。次に、決定された曲面形状を、決定された分割パターンによって分割し、曲面形状に沿った複数の面を生成する (ステップ S 4)。そして、生成された複数の面で構成される立体形状を表示する (ステップ S 5)。

## 明 細 書

立体形状表示プログラム、立体形状表示方法、および立体形状表示装置

## 5 技術分野

本発明は物の形状を立体的に表示するための立体形状表示プログラム、立体形状表示方法、および立体形状表示装置に関し、特に圧縮されたデータに基づいて元の形状を再現するための立体形状表示プログラム、立体形状表示方法、および立体形状表示装置に関する。

10

## 背景技術

近年、各種部品を3次元CAD(Computer-Aided Design)で立体的に表現することが一般的となっている。たとえば、複雑な構造の装置を構成する各部品が3次元CADデータで表現される。これにより、個々の部品のCADデータ（ソリッドモデル）を集結し、アセンブリ検討、レイアウト検討、干渉検討などの作業を、コンピュータ上で容易に行うことができる。このような作業はデジタルモックアップと呼ばれ、自動車や航空機産業を中心に盛んに用いられている。

一般にソリッドCADデータには、高精度の幾何と位相で表したプリサイズ表現(Precise Representation)が用いられる。ソリッドCADデータは、プリサイズ表現を基に、パラメトリック変形の為の拘束条件や形状生成履歴を持つ。さらにソリッドCADデータは、表示を速くするための表示データも保持している。それ故にソリッドCADが扱う情報量は多く、部品を集結してデジタルモックアップを造るにはデータ量が膨大になる。ソリッドのCADのデータ量が膨大であると、データの編集は行わずに立体形状を表示させるだけの場合であっても、膨大な量のデータ処理が発生する。

その解決策として自動車や航空機産業では、CADデータを表示用ポリゴンデータ（たとえば、VRML(Virtual Reality Modeling Language)のデータ）に変換し、その表示用ポリゴンデータをデジタルモックアップ専用のビューア(Viewer)を用いて表示している。表示用ポリゴンデータを用いて画面表示させる

ことで、形状の確認作業が容易となる。

ところが、最近のC A Dデータの詳細化（より細かい定義でデータ量が増加）は、ハードウェアやソフトウェアの進化を超え、専用ビューアでも製品全体を扱うことが困難になってきた。たとえば、自動車のエンジンの構成を表すC A Dデータ

5     データを、0.1mm の誤差で表示用ポリゴンデータに変換した場合、0. 5～1 Gbyte になる。エンジンだけでも0. 5～1 Gbyte になるため、車両1台分の合計は1 0 Gbyte 以上になると推定される。そのため、ビューアを実行するコンピュータには、表示用ポリゴンデータを格納するための大容量のメモリやハードディスク装置が必要となる。

10     そこで、表示用の立体形状のデータを圧縮する技術が考えられている。立体形状の圧縮表現としては B'ezier の三角パッチや Gregory パッチ（Lattice＝XVL）などを用いる方法があり、ハードディスク装置に格納しておくデータ容量を減らすことができるが、データをメモリ読み込み時にポリゴンに展開するためにメモリの削減にはならない。一方、圧縮された立体形状をそのままメモリに

15     保持し、表示倍率に応じてポリゴンを再分割する方法に Subdivision 法がある。Subdivision 法が適している理由は、画面の拡大時に多くの部品は視野（画面）外にある確率が高く、View Frustum Culling（視野外の物体を省く手法）を用いて省かれたC P U(Central Processing Unit)資源をポリゴンの再分割計算に割り当てることが出来るからである。

20     Subdivision 法の一例として、米国特許 6,222,553 号公報に記載された発明がある。この発明では、N U R B S(Non-Uniform Rational B-Splines)曲線などのコントロールポイントを用いて表現している。そして、1つの形状の中間点からその形状の各辺に向けて引いた線によって、元の形状を分割している。

しかし、従来の Subdivision 法は、メモリやファイル容量を少なくすることは

25     できるが、ポリゴンの再分割計算におけるコンピュータの処理負荷が大幅に増加してしまう。しかも、従来の Subdivision 法におけるデータ圧縮効果は、W e b などを用い遠隔地から大規模データのコラボレーションを行うには不十分である。

たとえば、米国特許 6,222,553 号公報に記載された発明では、ある形状の中間点から各辺に向けて引いた線でその形状を分割するため、分割する必要のない辺

（たとえば、直線の辺）が含まれていても全ての辺が分割されてしまう。従って、不必要な分割が発生し、処理負荷の増大を招いている。しかも、NURBS曲線などのコントロールポイントは、各辺毎に個別に設定する必要があり、データ量増大の原因となってしまう。

5

#### 発明の開示

本発明はこのような点に鑑みてなされたものであり、圧縮されたデータに基づいて高速に立体形状を表示することができる立体形状表示プログラム、立体形状表示方法、および立体形状表示装置を提供することを目的とする。

- 10 本発明の第1の形態では、上記課題を解決するために、図1に示す様な物の形状を立体的に表示させる処理をコンピュータに実行させる立体形状表示プログラムが提供される。本発明に係る立体形状表示プログラムを実行するコンピュータは、まず、少なくとも1つの面で立体形状を定義した立体形状データ2に基づいて、立体形状を構成する面の分割パターンを決定する（ステップS2）。同時に、
- 15 コンピュータは、立体形状データ2に予め定義された面の頂点の位置とその頂点における法線方向とに基づいて面の曲面形状を決定する（ステップS3）。次に、コンピュータは、決定された曲面形状を、決定された分割パターンによって分割し、曲面形状に沿った複数の面を生成する（ステップS4）。そして、コンピュータは、生成された複数の面で構成される立体形状を表示する（ステップS5）。
- 20 このような立体形状表示プログラムを任意のコンピュータで実行させることで、立体形状データ2を構成する面に定義された法線方向などに応じた曲面形状が決定され、その曲面形状が複数の面に分割される。そして、分割された面で構成される立体形状が表示される。

- 本発明の第2の形態では、物の形状をコンピュータで立体的に表示させるための
- 25 立体形状表示方法において、少なくとも1つの面で立体形状を定義した立体形状データに基づいて、前記立体形状を構成する前記面の分割パターンを決定すると共に、前記立体形状データに予め定義された前記面の頂点の位置と当該頂点における法線方向とに基づいて前記面の曲面形状を決定し、決定された前記曲面形状を、決定された前記分割パターンによって分割し、前記曲面形状に沿った複数

の面を生成し、生成された前記複数の面で構成される立体形状を表示する、ことを特徴とする立体形状表示方法が提供される。

このような立体形状表示方法によれば、立体形状データを構成する面に定義された法線方向などに応じた曲面形状が決定され、その曲面形状が複数の面に分割  
5 される。そして、分割された面で構成される立体形状が表示される。

本発明の第3の形態では、物の形状を立体的に表示する立体形状表示装置において、少なくとも1つの面で立体形状を定義した立体形状データに基づいて、前記立体形状を構成する前記面の分割パターンを決定する分割パターン決定手段と、前記立体形状データに予め定義された前記面の頂点の位置と当該頂点における法  
10 線方向とに基づいて、前記面の曲面形状を決定する曲面形状決定手段と、前記曲面形状決定手段で決定された前記曲面形状を、前記分割パターン決定手段で決定された前記分割パターンによって分割し、前記曲面形状に沿った複数の面を生成する面生成手段と、前記面生成手段で生成された前記複数の面で構成される立体形状を表示する表示手段と、を有することを特徴とする立体形状表示装置が提供  
15 される。

このような立体形状表示装置によれば、分割パターン決定手段により、少なくとも1つの面で立体形状を定義した立体形状データに基づいて、立体形状を構成する面の分割パターンが決定される。また、曲面形状決定手段により、立体形状データに予め定義された面の頂点の位置と当該頂点における法線方向とに基づい  
20 て、面の曲面形状が決定される。その後、面生成手段と、曲面形状決定手段で決定された曲面形状を、分割パターン決定手段で決定された分割パターンによって分割し、曲面形状に沿った複数の面が生成される。そして、表示手段により、面生成手段で生成された複数の面で構成される立体形状が表示される。

本発明の上記および他の目的、特徴および利点は本発明の例として好ましい実施  
25 施の形態を表す添付の図面と関連した以下の説明により明らかになるであろう。

#### 図面の簡単な説明

図1は、本実施の形態に適用される発明の概念図である。

図2は、本発明の実施の形態のシステム構成図である

図 3 は、本発明の実施の形態に用いるコンピュータのハードウェア構成例を示す図である。

図 4 は、本発明の実施の形態の機能を示す機能ブロック図である。

図 5 は、立体構造データ記憶部の内部構成を示す図である。

5 図 6 は、オブジェクト情報のデータ構造例を示す図である。

図 7 は、立体形状の一例を示す図である。図 7 (A) と図 7 (B) とは立体形状の斜視図である。

図 8 は、オブジェクト情報の具体例を示す図である。

図 9 は、立体形状表示手順を示す第 1 のフローチャートである。

10 図 10 は、立体形状表示手順を示す第 2 のフローチャートである。

図 11 は、立体形状表示手順を示す第 3 のフローチャートである。

図 12 は、空間曲線のポリライン近似の例を示す図である。

図 13 は、図形の分割例を示す図である。

図 14 は、3 辺形の分割パターンの種別を示す図である。

15 図 15 は、3 辺形分割パターン決定処理の手順を示すフローチャートである。

図 16 は、4 辺形の分割パターンの種別を示す図である。

図 17 は、4 辺形分割パターン決定処理の手順を示すフローチャートである。

図 18 は、隣接する面同士が滑らかに接続している立体形状の一例を示す図である。

20 図 19 は、2 次微分ベクトルを用いて決定されたパラメータ 3 次式の一例を示す図である。

図 20 は、隣接する面同士が共有する辺で折れている立体形状の一例を示す図である。

図 21 は、辺の形状を強制的に指示された立体形状の一例を示す図である。

25 図 22 は、分割対象の面の一例を示す図である。

図 23 は、分割される 3 辺形を含む立体形状の一例を示す図である。

図 24 は、分割される 3 辺形の一例を示す図である。

図 25 は、分割される 4 辺形の一例を示す図である。

図 26 は、ハイブリッドメッシュデータで定義されている立体形状の辺を表示

した画面例を示す図である。

図 27 は、ハイブリッドメッシュデータで定義されている立体形状を表示した画面例を示す図である。

図 28 は、ハイブリッドメッシュデータを再分割したときの立体形状の辺を表示した画面例を示す図である。

図 29 は、ハイブリッドメッシュデータを再分割したときの立体形状を表示した画面例を示す図である。

図 30 は、自動車の立体形状の表示画面例を示す図である。

## 10 発明を実施するための最良の形態

以下、本発明の実施の形態を図面を参照して説明する。

まず、本発明の実施の形態に適用される発明の概要について説明し、その後、本発明の実施の形態の具体的な内容を説明する。

図 1 は、本実施の形態に適用される発明の概念図である。本発明は、物の形状を立体的に表示させる処理をコンピュータに実行させる立体形状表示プログラムである。この立体形状表示プログラムに基づいて、コンピュータが図 1 に示すような処理を実行する。

コンピュータは、CADデータ 1 のデータ圧縮を行い、圧縮された立体形状データ 2 を生成する。立体形状データ 2 は、CADデータ 1 における形状を定義するデータを圧縮したものである。データ圧縮により、たとえば、複雑な曲面形状が、単純な形状の複数の面に置き換えられる。

その後、コンピュータは、立体形状データ 2 に基づいて、立体形状を構成する面の分割パターンを決定する（ステップ S 2）。たとえば、面を構成する辺の分割数を決定し、各辺の分割数に応じた面の分割パターンを決定する。

また、コンピュータは、立体形状データ 2 に予め定義された面の頂点の位置とその頂点における法線方向とに基づいて、面の曲面形状を決定する（ステップ S 3）。たとえば、法線方向に垂直な方向が曲面形状の接線となるように、曲面形状を決定する。なお、法線方向は、一般には、法線ベクトル（単位ベクトル）で定義される。



次に、コンピュータは、決定された曲面形状を、決定された分割パターンによって分割し、曲面形状に沿った複数の面を生成する（ステップS4）。そして、コンピュータは、生成された複数の面で構成される立体形状を表示する（ステップS5）。

- 5      このような立体形状表示プログラムを任意のコンピュータで実行させることで、立体形状データ2を構成する面に定義された法線方向などに応じた曲面形状が決定され、その曲面形状が複数の面に分割される。そして、分割された面で構成される立体形状が表示される。

- 10      これにより、元のCADデータ1で定義された立体形状を、少ないデータ量の立体形状データ2で記録することができる。そして、立体形状データ2に基づいて、CADデータ1で定義された形状を少ない誤差で再現することが可能となる。

- 15      しかも、立体形状データ2では、立体形状を構成する面の頂点座標と法線方向が設定されている。法線方向は、たとえば、法線ベクトルで定義される。この法線ベクトルは、リアルな画面表示を行うためのグーローシェーディングや光の反射などの計算に必要な情報である。本発明では、立体形状データ2の状態では法線方向が定義されているため、立体形状データ2自体を1つのポリゴンモデルとみなして、そのままリアルに（たとえば、グーローシェーディングを用いる）画面表示することができる。従って、状況に応じて、コンピュータが面の分割の要否を判断し、面の分割を行わない高速表示と、面を分割する高精度の表示とを適宜  
20      切り換えることが可能である。

- 25      なお、前述した米国特許 6,222,553 号公報記載の発明に代表されるように、従来の Subdivision 法で圧縮された多くのデータは、法線方向の情報（法線ベクトル）を面の分割に利用していない。そのため、圧縮されたデータに法線ベクトルを含めれば、面の分割に不要なデータを余分にもつことになり、データ容量が増加してしまう。

また、法線方向によって曲面形状を決定することで、元の立体形状を再現するために必要なデータ量を少なく抑えることができる。すなわち、滑らかに接続される2つの面では、ほとんどの場合、その境界の辺の端点において、法線方向の定義情報（法線ベクトル）を共有することができる。したがって、立体形状デー



圧縮されたデータを各コンピュータ 100, 100a, 100b間で受け渡すことで、CADを用いた共同設計作業（コラボレーション）を行うことができる。

以下、コンピュータ 100によって本発明を実施する場合の形態について詳しく説明する。

- 5 図3は、本発明の実施の形態に用いるコンピュータのハードウェア構成例を示す図である。コンピュータ 100は、CPU 101によって装置全体が制御されている。CPU 101には、バス 107を介してRAM(Random Access Memory) 102、ハードディスクドライブ（HDD:Hard Disk Drive） 103、グラフィック処理装置 104、入力インタフェース 105、および通信インタフェース 106が接続されている。

RAM 102には、CPU 101に実行させるOS(Operating System)のプログラムやアプリケーションプログラムの少なくとも一部が一時的に格納される。また、RAM 102には、CPU 101による処理に必要な各種データが格納される。HDD 103には、OSやアプリケーションプログラムが格納される。

- 15 グラフィック処理装置 104には、3次元表示回路を内蔵しており、モニタ 11が接続されている。グラフィック処理装置 104は、CPU 101からの命令に従って、3次元の座標データ（ポリゴンデータなど）に基づく画像を生成し、その画像をモニタ 11の画面に表示させる。入力インタフェース 105には、キーボード 12とマウス 13とが接続されている。入力インタフェース 105は、  
20 キーボード 12やマウス 13から送られてくる信号を、バス 107を介してCPU 101に送信する。

通信インタフェース 106は、ネットワーク 10に接続されている。通信インタフェース 106は、ネットワーク 10を介して、他のコンピュータとの間でデータの送受信を行う。

- 25 以上のようなハードウェア構成によって、本実施の形態の処理機能を実現することができる。

図4は、本発明の実施の形態の機能を示す機能ブロック図である。コンピュータ 100は、ビューア 110、ビューアアプリケーション部 130、3次元表示用API(Application Program Interface) 140、OS API 150を有してい

る。

ビューア 110 は、立体形状を表すデータのデータ形式を変換し、データ容量を圧縮する。たとえば、VRML データ 31 や CAD データ 32 を、ハイブリッドメッシュデータに変換する。また、ビューア 110 は、ハイブリッドメッシュ  
5 データに基づいて、立体形状を所定の誤差で再現する。再現される立体形状は、複数のポリゴンで構成されるポリゴンモデルである。ビューア 110 内の構成の詳細は後述する。

ビューアアプリケーション部 130 は、ビューア 110 と連携して CAD を用いた設計作業を支援するための付加機能を有している。図 4 の例では、ビューア  
10 アプリケーション部 130 は、付加機能として干渉検討部 131、計測部 132、分解図生成部 133、イラスト生成部 134 などをも有している。干渉検討部 131 は、ビューア 110 内から渡された立体形状のデータに基づいて、部品間の干渉の有無を判断する。計測部 132 は、ビューア 110 内から渡された立体形状のデータに基づいて、任意の 2 点間の距離を計算する。分解図生成部 133 は、  
15 ビューア 110 内から渡された立体形状のデータに基づいて、立体形状の分解図を生成する。イラスト生成部 134 は、ビューア 110 内から渡された立体形状のデータに基づいて、立体形状のイラスト画像を生成する。

3次元表示用 API 140 は、ビューア 110 などの各種処理機能と 3次元表示を行うためのグラフィック処理装置 104 との間のインタフェースである。具  
20 体的には、3次元表示用 API 140 は、ビューア 110 から渡されたポリゴンなどの情報に基づいて、コンピュータ 100 における 3次元表示処理要求を生成し、その 3次元表示処理要求をコンピュータ 100 に実行させる。たとえば、3次元表示用 API 140 は、ビューア 110 から渡されたポリゴンの端点座標や色などの情報を、表示命令と共に図 3 に示すグラフィック処理装置 104 に渡す。  
25 すると、グラフィック処理装置 104 が各ポリゴンの透視投影などを行い、複数のポリゴンで構成される立体形状を表す画像（フレーム画像）を生成する。そして、グラフィック処理装置 104 で生成された画像のデータが所定周期でモニタ 11 に送信され、立体形状を表す画像がモニタ 11 の画面に表示される。3次元表示用 API 140 としては、たとえば、OpenGL（商標）を用いることが

できる。

OSAPI150は、OSと他の処理機能との間のインタフェースである。具体的には、OSAPI150は、ビューア110などから出力される処理要求に対応するOSの処理機能を起動し、コンピュータ100に処理を実行させる。

- 5      次に、ビューア110の内部構成について説明する。

ビューア110は、アプリケーションインタフェース111、ハイブリッドメッシュAPI112、VRMLデータ変換部113、CADデータ変換部114、ユーザインタフェース115、再分割部116、表示データ出力部117、および、立体構造データ記憶部120を有している。

- 10      アプリケーションインタフェース111は、ビューアアプリケーション部130との間のインタフェースである。具体的には、アプリケーションインタフェース111は、ハイブリッドメッシュAPI112を介して受け取った立体構造データ（モックアップデータ）を、ビューアアプリケーション部130内の干渉検討部131などに渡す。また、アプリケーションインタフェース111は、ビューアアプリケーション部130内の干渉検討部131などから受け取った立体構造データをハイブリッドメッシュAPI112に渡す。
- 15

- ハイブリッドメッシュAPI112は、アプリケーションインタフェース111に対して立体構造データ記憶部120へのアクセス環境を提供するインタフェースである。具体的には、ハイブリッドメッシュAPI112は、アプリケーションインタフェース111を介した要求に応じて、立体構造データ記憶部120内のハイブリッドメッシュデータの一部（または全部）のデータをアプリケーションインタフェース111に渡す。また、ハイブリッドメッシュAPI112は、アプリケーションインタフェース111を介して受け取ったデータを、ハイブリッドメッシュデータの一部（または全部）として立体構造データ記憶部120に格納する。
- 20
- 25

VRMLデータ変換部113は、VRMLデータ31をハイブリッドメッシュデータに変換する。この変換により、立体形状を表すデータのデータ容量が小さくなる。そして、VRMLデータ変換部113は、変換後のハイブリッドメッシュデータを、立体構造データ記憶部120に格納する。

CADデータ変換部114は、CADデータ32をハイブリッドメッシュデータに変換する。この変換により、立体形状を表すデータのデータ容量が小さくなる。そして、CADデータ変換部114は、変換後のハイブリッドメッシュデータを、立体構造データ記憶部120に格納する。

- 5 ユーザインタフェース115は、OSAPI150と連携し、キーボード12やマウス13を用いた操作入力の情報を取得する。そして、ユーザインタフェース115は、操作入力に応答して、ビューア110内の各構成要素に対して、処理要求を出したり、データを渡したりする。たとえば、ユーザインタフェース115は、操作入力によって指定された表示倍率116aや許容誤差116bを再分割部116に渡す。また、ユーザインタフェース115は、表示対象のオブジェクトを投影するための視点と立体形状との相対的位置の移動指示が入力された場合（表示画面内で立体形状が動く場合）、移動中であることを再分割部116に通知する。

- 再分割部116は、ハイブリッドメッシュデータを再分割する。すなわち、再分割部116は、CADデータ32よりも粗い構造で表現されたハイブリッドメッシュデータを構成する面（フェース）を分割し、元のCADデータ32の形状に近づける。そして、再分割部116は、面を分割する毎に、分割後の面の情報を表示データ出力部117に渡す。

- 20 なお、再分割部116は、ハイブリッドメッシュデータの面の分割数を決定するためのパラメータとして、表示倍率116aと許容誤差116bとを保持している。表示倍率116aは、ハイブリッドメッシュデータで定義される立体形状の画面表示上での表示倍率である。許容誤差116bは、ハイブリッドメッシュデータに基づいて定義される立体形状（曲面で構成される形状）を平面のポリゴンで表した場合に発生する形状誤差の最大値である。再分割部116は、表示倍率116aが大きいほどポリゴンの分割数を多くし、表示倍率116aが小さいほどポリゴンの分割数を少なくする。また、再分割部116は、許容誤差116bが大きいほどポリゴンの分割数を少なくし、許容誤差116bが小さいほどポリゴンの分割数を多くする。

表示データ出力部117は、再分割部116で分割された後のポリゴンの情報

を、3次元表示用API140に対して出力する。

立体構造データ記憶部120は、表示対象のモデル（たとえば、自動車）の立体構造（部品の立体形状の組み合わせ）のデータを記憶するためのRAM102内の記憶領域である。立体構造のデータは、主に、表示対象モデルの構成要素（たとえば、自動車の部品）毎の立体形状を示すハイブリッドメッシュデータで形成される。このハイブリッドメッシュデータは、VRMLデータ変換部113やCADデータ変換部114によって生成される。また、ハイブリッドメッシュデータは、HDD103内のハイブリッドメッシュファイル33から読み込まれる場合もある。

- 10     ハイブリッドメッシュデータには、立体形状を構成する面毎の形状情報が定義されており、面単位の情報をポリゴンとみなして、そのまま3次元表示用API140に渡して画面表示させることができる。また、ハイブリッドメッシュデータには、各面を分割するのに必要な情報（基底分割数など）が含まれる。

次に、立体構造データ記憶部120のデータ構造の詳細について説明する。

- 15     図5は、立体構造データ記憶部の内部構成を示す図である。立体構造データ記憶部120は、複数のハイブリッドメッシュデータ200、200a、200b、・・・とオブジェクト配置情報300とを有している。ハイブリッドメッシュデータ200、200a、200b、・・・は、表示対象物を構成する部品（オブジェクト）毎の形状を定義したデータである。ハイブリッドメッシュデータ200、200a、200b、・・・は、表示対象物を構成するユニット毎にグループ化されている場合もある。ここで、ユニットとは、1以上の部品で構成される物品である。表示対象物をアセンブリ（組み立て部品）とすれば、ユニットはサブアセンブリである。たとえば、表示対象物が自動車であれば、エンジンやサスペンションなどがユニットに相当する。なお、1つのユニットがさらに複数のユニット（各ユニットが1以上の部品を有する）で構成される場合もある。

25     ハイブリッドメッシュデータ200には、オブジェクト情報210、点座標データ群220、およびベクトルデータ群230が含まれている。オブジェクト情報210は、各部品の立体形状などを表すデータ（部品の色の情報も含む）である。オブジェクト情報210の詳細は後述する。

点座標データ群 2 2 0 は、立体形状を構成する面の各端点の端点座標データ 2 2 1, 2 2 2, ..., 2 2 i, ..., 2 2 m, 2 2 n の集合である。端点座標データ 2 2 1, 2 2 2, ..., 2 2 i, ..., 2 2 m, 2 2 n には、x 軸-y 軸-z 軸で表される 3 次元空間（ローカル座標系）内での各軸方向の座標値が設定されている。

ベクトルデータ群 2 3 0 は、立体形状を構成する面の各端点に定義する法線ベクトルまたは接線ベクトルを表すベクトルデータ 2 3 1, 2 3 2, ..., 2 3 i, ..., 2 3 m, 2 3 n の集合である。ベクトルデータ 2 3 1, 2 3 2, ..., 2 3 i, ..., 2 3 m, 2 3 n は、3 次元空間（ローカル座標系）内での方向と大きさを、x 軸、y 軸、z 軸方向のベクトル成分で表している。

なお、本実施の形態において法線ベクトルとして指定されるベクトルは、全て長さが 1 の単位ベクトルである。接線ベクトルとして指定されるベクトルは、単位ベクトルとは限らない。

オブジェクト配置情報 3 0 0 は、立体形状をグローバル座標系（ワールド座標系）に配置する場合の配置位置および向きを定義した情報である。各立体形状に関する配置情報が定義されることで、表示対象物を構成する各部品をグローバル座標系内に配置し、複数の立体形状の組み合わせからなる表示対象物の形を生成することができる。

次に、オブジェクト情報の内容を詳細に説明する。

図 6 は、オブジェクト情報のデータ構造例を示す図である。なお、図 6 中のデータ名の前に示されている矢印（↑）は、そのデータが他のデータの格納位置を指すポインタであることを示している。

オブジェクト情報 2 1 0 は、オブジェクト識別情報(ObjectID) 2 1 1、最大基底分割数(MaxDivide) 2 1 2、および複数のフェース情報(Face1,...,Face i,...,Face n-1,Face n) 4 1 0, 4 2 0, 4 3 0, 4 4 0 を有している。オブジェクト識別情報 2 1 1 は、表示対象物を構成する部品ごとに作成されたオブジェクト情報 2 1 0 を一意に識別するための識別情報である。最大基底分割数 2 1 2 は、オブジェクト情報 2 1 0 を構成する面の各辺に設定された基底分割数のうちの最大値である。フェース情報 4 1 0, 4 2 0, 4 3 0, 4 4 0 は、部品を構成する



面の情報である。

フェース情報 4 2 0 は、フェース識別情報(Face ID) 4 2 1、フェース辺数 (nP) 4 2 2、フェース属性(Attr) 4 2 3、複数の端点情報(Vertex 1,Vertex 2,...,Vertex i,...,Vertex nP-1,Vertex nP) 4 2 4, 4 2 5, 4 2 6, 4 2 7, 4 2 8、および再帰ポインタ(↑ EOF) 4 2 9 を有している。フェース識別情報 4 2 1 は、オブジェクト情報 2 1 0 内でフェース情報を一意に識別するための識別情報である。フェース辺数 4 2 2 は、面を構成する辺（直線または曲線）の数である。フェース属性 4 2 3 は、面の色などの属性情報である。

端点情報 4 2 4, 4 2 5, 4 2 6, 4 2 7, 4 2 8 は、面の端点に関する情報である。再帰ポインタ 4 2 9 は、先頭の端点情報 4 2 4 の位置を指し示すポインタである。再帰ポインタ 4 2 9 により先頭の端点情報 4 2 4 が示されることで、端点情報 4 2 4, 4 2 5, 4 2 6, 4 2 7, 4 2 8 は、端点の列び順のループを構成する。連続した 2 つの端点情報によって、面を構成する辺が定義されている。なお、最後尾の端点情報 4 2 8 は、先頭の端点情報 4 2 4 と連続しているものとして扱われる。

端点情報 4 2 6 は、点座標ポインタ(↑ Point) 4 2 6 a、法線ベクトルポインタ(↑ Normal) 4 2 6 b、接線定義情報(Ratio or Tangent) 4 2 6 c、基底分割数(nDevide) 4 2 6 d、および隣接辺ポインタ(↑ MateEdge) 4 2 6 e を有している。

点座標ポインタ 4 2 6 a は、点座標データ群 2 2 0 内の点座標データ 2 2 1, 2 2 2, ..., 2 2 i, ..., 2 2 m, 2 2 n のうち、端点の 3 次元空間（ローカル座標系）内での位置を示す点座標データを指すポインタである。法線ベクトルポインタ 4 2 6 b は、ベクトルデータ群 2 3 0 内のベクトルデータ 2 3 1, 2 3 2, ..., 2 3 i, ..., 2 3 m, 2 3 n のうち、端点の法線ベクトルの値を示すベクトルデータを指すポインタである。

接線定義情報 4 2 6 c は、端点における接線ベクトルを定義するための情報である。接線定義情報 4 2 6 c には、端点における接線の長さ(Ratio)、または接線ベクトルを指すポインタ(Tangent)のいずれかが設定される。法線ベクトルと辺の両端の端点の位置とから接線ベクトルの方向が決定できる場合（辺が法線ベ

クトルの示す方向に湾曲する場合)には、接線定義情報4 2 6 cとして接線の長さが設定される。法線ベクトルと辺の両端の端点の位置とから接線ベクトルの方向が決定できない場合(辺が法線ベクトルの示す方向と異なる方向に湾曲する場合)には、接線定義情報4 2 6 cとして接線ベクトルへのポイントが設定される。

- 5 基底分割数4 2 6 dは、予め設定された辺の分割数である。基底分割数4 2 6 dは、表示倍率が1(拡大も縮小も行わない)のときに、分割後の誤差が予め決められた誤差以内となるような値が設定される。

- 隣接辺ポイント4 2 6 eは、隣接する面と辺を共有する場合に、隣接する面側の共有する辺の情報(辺の両端を示す2つの端点情報の先頭の位置)を指し示す  
10 ポイントである。隣接辺ポイント4 2 6 eにより、隣接する面を繋ぎ合わせたループ間での面同士の辺の供給関係が定義される。

なお、複数の辺の端点情報4 2 6において、点座標ポイント4 2 6 aが同じアドレス(点座標データの位置)を示す場合、各辺の端点が共有されることを表している。

- 15 また、複数の辺の端点が共有されるとき、それぞれの辺の端点に対応する端点情報4 2 6の法線ベクトルポイント4 2 6 bが同じアドレス(法線ベクトルデータの位置)を示していれば、各辺の法線ベクトルが共有されている。

- このように、本実施の形態のハイブリッドメッシュデータ2 0 0, 2 0 0 a, 2 0 0 bでは、端点のループと面のループとが定義されていると共に、辺の端点  
20 の共有関係、法線ベクトルの共有関係、辺の共有関係が定義されている。これにより、立体形状の位相(topology)と幾何(geometry)との情報が定義される。ここで、位相とは、面、稜線、端点(頂点)など立体の境界面のつながり具合をあらわす要素であり、幾何とは、曲面、曲線、点など立体の実際の形状をあらわす要素である。

- 25 次に、図7～図8を参照して、立体形状に応じたオブジェクト情報2 1 0の具体例を説明する。

図7は、立体形状の一例を示す図である。図7(A)と図7(B)とは立体形状の斜視図である。図7(A)と図7(B)とは、見る方向が逆である。

立体形状5 0 0は、6つの面5 1 1～5 1 6で構成され、1 2個の辺5 2 1～

5 3 2、8個の端点5 4 1～5 4 8を有している。各面5 1 1～5 1 6のフェース識別情報(FaceID)は、F 1～F 6である。面5 1 1は曲面であり、他の面5 1 2～5 1 6は平面である。

ここで、各端点5 4 1～5 4 8の位置を示す点座標データへのポインタ（点座  
5 標ポインタ）をP 1～P 8とし、各端点5 4 1～5 4 8の法線ベクトルを示すベ  
クトルデータへのポインタ（法線ベクトルポインタ）をN 1～N 8とする。なお、  
図7に示すように端点5 4 1には、2つの法線ベクトル5 5 1、5 5 2が存在す  
る。このとき、法線ベクトル5 5 1、5 5 2を有する面のフェース識別情報を、  
法線ベクトルデータを示す記号の後に括弧書きで示す。すなわち、面5 1 6が有  
10 する法線ベクトル5 5 1の法線ベクトルポインタはN 1（F 6）であり、面5 1  
1と面5 1 3が共有する法線ベクトル5 5 2の法線ベクトルポインタはN 1（F  
1、F 3）である。

このような立体形状5 0 0のオブジェクト情報の例を以下に示す。

図8は、オブジェクト情報の具体例を示す図である。オブジェクト情報2 1 0  
15 aには、オブジェクト識別情報2 1 1 a、最大基底分割数2 1 2 a、および複数の  
フェース情報4 1 0 a、4 2 0 a、4 3 0 a、4 4 0 a、4 5 0 a、4 6 0 a  
が含まれている。フェース情報4 1 0 aは、フェース識別情報「F 1」の面5 1  
1に関する情報である。フェース情報4 2 0 aは、フェース識別情報「F 2」の  
面5 1 2に関する情報である。フェース情報4 3 0 aは、フェース識別情報「F  
20 3」の面5 1 3に関する情報である。フェース情報4 4 0 aは、フェース識別情  
報「F 4」の面5 1 4に関する情報である。フェース情報4 5 0 aは、フェース  
識別情報「F 5」の面5 1 5に関する情報である。フェース情報4 6 0 aは、フ  
ェース識別情報「F 6」の面5 1 6に関する情報である。

ここで、フェース情報4 1 0 aを参照して、データの例を詳しく説明する。フ  
25 ェース情報4 1 0 aのフェース識別情報(FaceID)は「F 1」である。これによ  
り、図7に示す面5 1 1に関する情報であることが分かる。また、フェース辺数  
(nP)は、4である。これにより、4辺からなる面であることが分かる。フェース  
情報(Attr)には、色情報などが設定されている。

フェース情報4 1 0 aには、4つの端点5 4 1～5 4 4に対応する端点情報が

設定されている。図 8 の例では、端点 5 4 1 の端点情報には、点座標ポインタ P 1、法線ベクトルポインタ N 1 (F 1, F 3)、接線の長さ(Ratio)、基底分割数「0」、および面 5 1 3 (フェース識別情報 F 3) の先頭の端点情報 (F 3 - 1) を指し示す隣接辺ポインタが設定されている。端点 5 4 2 の端点情報には、  
5 点座標ポインタ P 2、法線ベクトルポインタ N 2 (F 1, F 3)、接線の長さ (Ratio)、基底分割数「n 0 - 1」、および面 5 1 4 (フェース識別情報 F 4) の 4 番目の端点情報 (F 4 - 4) を指し示す隣接辺ポインタが設定されている。端点 5 4 3 の端点情報には、点座標ポインタ P 3、法線ベクトルポインタ N 3 (F 1)、接線の長さ(Ratio)、基底分割数「0」、および面 5 1 5 (フェース識別情報 F 5) の 4 番目の端点情報 (F 5 - 4) を指し示す隣接辺ポインタが設定され  
10 ている。端点 5 4 4 の端点情報には、点座標ポインタ P 4、法線ベクトルポインタ N 4 (F 1)、接線の長さ(Ratio)、基底分割数「n 0 - 1」、および面 5 1 6 (フェース識別情報 F 6) の先頭の端点情報 (F 6 - 1) を指し示す隣接辺ポインタが設定されている。再帰ポインタは、先頭の端点情報を指し示している。  
15 以上のような機能構成およびデータ構造を有するコンピュータ 1 0 0 によって、立体形状の表示が行われる。以下、立体形状の表示方法について詳しく説明する。

図 9 は、立体形状表示手順を示す第 1 のフローチャートである。以下、図 9 に示す処理をステップ番号に沿って説明する。

[ステップ S 1 1] コンピュータ 1 0 0 は、ユーザからの操作入力などによる  
20 指示に従って、ハイブリッドメッシュデータで構成される立体形状データを生成する。たとえば、VRML データ変換部 1 1 3 が、VRML データ 3 1 に定義されている部品の立体形状を解析し、部品毎の立体形状データをハイブリッドメッシュデータに変換する。また、CAD データ変換部 1 1 4 が、CAD データ 3 2 に定義されている部品の立体形状を解析し、部品毎の立体形状データをハイブリ  
25 ッドメッシュデータに変換する。

生成されるハイブリッドメッシュデータは、元の VRML データ 3 1 に定義された立体形状よりも面の数を減らした立体形状を定義している。したがって、この変換処理により、データ容量が圧縮される (データ容量が減少する)。

[ステップ S 1 2] VRML データ変換部 1 1 3 または CAD データ変換部 1

14は、データ変換によって生成したハイブリッドメッシュデータを内部のRAM102内の立体構造データ記憶部120に格納する。なお、RAM102に格納されたハイブリッドメッシュデータは、ハイブリッドメッシュファイル33に書き出すことができる。また、ハイブリッドメッシュファイル33からハイブリッドメッシュデータを取り出し、RAM102内の立体構造データ記憶部120に格納することもできる。

[ステップS13] ユーザインタフェース115は、ユーザからのキーボード12などに対する操作入力に基づいて、画面の表示倍率と許容誤差を取得する。すると、ユーザインタフェース115は、再分割部116に対して表示倍率と許容誤差を渡す。

[ステップS14] 再分割部116は、立体構造データ記憶部120内のオブジェクトを1つ選択する。以下、選択したオブジェクトに関して、ステップS15～ステップS37の処理が行われる。

[ステップS15] 再分割部116は、立体構造データ記憶部120からステップS14で選択したオブジェクトのハイブリッドメッシュデータを取り出し、最大基底分割数を取得する。

[ステップS16] 再分割部116は、画面の表示倍率、許容誤差、および最大基底分割数に基づいて、実分割数を算出する。実分割数の算出方法の詳細については後述する。

[ステップS17] 再分割部116は、分割処理の必要の有無を判断する。具体的には、ステップS16で算出した実分割数が1以下か否かを判断する。実分割数が1以下の場合、分割処理の必要がないと判断される。一方、実分割数が1を超えた場合には、分割処理の必要があると判断する。

すなわち、最大基底分割数は、単位倍率のときの単位誤差における分割数が最も大きい辺の基底分割数である。したがって、その辺の分割が不要であれば、他の辺の分割も不要である。

分割処理の必要がない場合には、処理がステップS18に進められる。分割処理が必要な場合には、処理が図10のステップS21に進められる。

[ステップS18] 再分割部116は、立体構造データ記憶部120に記憶さ

れたハイブリッドメッシュデータやオブジェクト配置情報を、そのまま表示データ出力部 1 1 7 に送信する。表示データ出力部 1 1 7 は、ハイブリッドメッシュデータを 3 次元表示用 API 1 4 0 へ出力する。すなわち、ハイブリッドメッシュデータは、そのままポリゴンモデルのデータとして解釈することができるため、  
5 そのままのデータ形式で 3 次元表示用 API 1 4 0 に出力すれば、少ない数の面（ハイブリッドメッシュデータで定義されている面の数）で構成された立体形状 4 1 がモニタ 1 1 の画面の表示される。その後、処理がステップ S 3 8 に進められる。

これにより、表示倍率が小さい場合（画面中で小さく表示される場合）には、  
10 簡易な構造の立体形状が表示され、高速に画面表示が可能となる。また、許容誤差が大きい場合にも簡易な構造の立体形状が表示され、高速に画面表示が可能となる。

図 1 0 は、立体形状表示手順を示す第 2 のフローチャートである。以下、図 1 0 に示す処理をステップ番号に沿って説明する。

15 [ステップ S 2 1] 再分割部 1 1 6 は、立体形状の分割処理に必要なパラメータ等の初期化処理を行う。

[ステップ S 2 2] 再分割部 1 1 6 は、立体構造データ記憶部 1 2 0 に格納されているハイブリッドメッシュデータの中から、フェース情報を 1 つ選択する。

20 [ステップ S 2 3] 再分割部 1 1 6 は、視点（仮想カメラ）と立体形状（ステップ S 2 2 で選択した面）との相対位置が変化しているか否かを判断する。すなわち、表示される画面内で立体形状が動くか否かを判断する。相対位置が変化している場合には、処理がステップ S 2 9 に進められる。相対位置が変化していない場合には、処理がステップ S 2 4 に進められる。なお、本実施の形態では、視点と立体形状との相対位置は、ユーザからの操作入力（視点の移動指示や、視点  
25 の向き（視軸）の変更指示など）に応じて変化する。

[ステップ S 2 4] 再分割部 1 1 6 は、選択したフェース情報で定義される辺（端点情報の列）の 1 つを選択する。

[ステップ S 2 5] 再分割部 1 1 6 は、選択した辺の実分割数を算出する。これは、選択された辺の基底分割数  $n_0$  に、1 つの画面内に表示される全ての立体

形状に対して共通の値 $\sqrt{\quad}$  (Scale/Tol) を乗算したものである。

〔ステップS 2 6〕再分割部1 1 6は、ステップS 2 5で算出された実分割数を、2のべき乗に切り上げる。辺の分割数を2のべき乗とすることで、以降の3角形ポリゴンの分割を、単純な分割パターンで行うことができる。

- 5      〔ステップS 2 7〕再分割部1 1 6は、ステップS 2 2で選択したフェース情報の全ての辺の分割数が決定したか否かを判断する。全ての辺の分割数が決定した場合には、処理がステップS 2 8に進められる。未処理の辺があれば、処理がステップS 2 4に進められる。

- 10     〔ステップS 2 8〕再分割部1 1 6は、全ての辺で分割の必要がないか（分割数が1以下か）否かを判断する。全ての辺で分割の必要がなければ、処理がステップS 2 9に進められる。分割の必要な辺が1つでもあれば、処理が図1 1のステップS 3 1に進められる。

- 15     〔ステップS 2 9〕再分割部1 1 6は、立体構造データ記憶部1 2 0に記憶されたハイブリッドメッシュデータのステップS 2 2で選択した面に関する情報を、そのまま表示データ出力部1 1 7に送信する。表示データ出力部1 1 7は、ハイブリッドメッシュデータを3次元表示用API 1 4 0へ出力する。すると、ステップS 2 2で選択された面4 0が、再分割されずにモニタ1 1に表示される。その後、処理は図1 1のステップS 3 7に進む。

- 20     このように、画面中で立体形状が動いている場合には、各面が分割されずに画面表示される。その結果、少ない数の面（ハイブリッドメッシュデータで定義されている面の数）で構成された立体形状がモニタ1 1の画面の表示される。すなわち、少ない処理で高速に立体形状の画面表示が可能となる。

図1 1は、立体形状表示手順を示す第3のフローチャートである。以下、図1 1に示す処理をステップ番号に沿って説明する。

- 25     〔ステップS 3 1〕再分割部1 1 6は、図1 0のステップS 2 2で選択したフェース情報が3辺形か4辺形かを判断する。3辺形の場合には、処理がステップS 3 2に進められる。4辺形の場合には、処理がステップS 3 3に進められる。

〔ステップS 3 2〕再分割部1 1 6は、3辺形分割パターン決定処理を行う。この処理は、各辺の実分割数に応じて、3辺形を1段階分だけ分割パターンを決

定する処理である。すなわち、3辺形の分割処理は、多段階に分けて行われる。ステップS 3 2では、その1段階分の分割パターンの決定が行われる。なお、3辺形分割処理の詳細は後述する。その後、処理がステップS 3 4に進められる。

〔ステップS 3 3〕再分割部1 1 6は、4辺形分割パターン決定処理を行う。

- 5 この処理は、各辺の実分割数に応じて、4辺形を1段階分だけ分割パターンを決定する処理である。すなわち、4辺形の分割処理は、多段階に分けて行われる。ステップS 3 3では、その1段階分の分割パターンの決定が行われる。この処理の詳細は後述する。

〔ステップS 3 4〕再分割部1 1 6は、各辺のパラメータ3次式を決定する。

- 10 〔ステップS 3 5〕再分割部1 1 6は、ステップS 3 2、S 3 3で決定した分割パターンとステップS 3 4で決定したパラメータ3次式とに従って、立体形状を構成する各面を再分割する。再分割された結果が、これ以上の再々分割できない場合、ステップS 3 6へ進む。

- 15 再々分割できる場合は、再分割部1 1 6は、再帰的にステップS 3 1～S 3 6の処理を呼び出す。再帰的に呼び出されたときのステップS 3 2、S 3 3では、多段階に分けて行われる再分割処理の次の1段階分の分割パターン決定処理が行われる。このように、ステップS 3 1～S 3 6の処理を再帰的に呼び出すことで、ステップS 2 2で選択した面が、段階的に細分化される。

- 20 〔ステップS 3 6〕再分割部1 1 6は、再分割した後の立体形状データを、表示データ出力部1 1 7に送信する。表示データ出力部1 1 7は、立体形状データを3次元表示用API 1 4 0へ出力する。これにより、再分割の分割数に応じた精度の立体形状4 2～4 4がモニタ1 1の画面に表示される。

- 25 なお、ステップS 3 6の処理は、選択された面がこれ以上分割できない状態まで分割されたとき実行される。そして、最小単位の3辺形が生成される毎に、その3辺形の表示用のポリゴンデータが表示データ出力部1 1 7に送信される。そのため、立体形状全体のポリゴンデータを、再分割部1 1 6が保持する必要はない。

〔ステップS 3 7〕再分割部1 1 6は、全ての面（フェース）を分割したか否かを判断する。全ての面を分割した場合には、処理がステップS 3 8に進められ



る。未分割の面がある場合には、処理がステップS 2 2に進められる。

[ステップS 3 8] 再分割部1 1 6は、全オブジェクトの画面表示が終了したか否かを判断する。なお、ユーザの操作入力によりビューA 1 1 0の終了が指示された場合にも、画面表示終了と判断される。画面表示終了の場合には処理が終了し、別のオブジェクトのハイブリッドメッシュデータの画面表示を継続する場合

5 合には、処理が図9のステップS 1 4に進められる。

以上のようにして、拡大倍率、許容誤差、および視点との相対位置に応じた精度の立体形状が画面に表示される。しかも、粗い形状の面を定義しているハイブリッドメッシュデータに基づいて、精度の高い（誤差の少ない）立体形状を表示

10 することができる。そのため、立体形状データを格納するための記憶領域が少なくてすむ。

以下、図9～図11で説明したステップ毎の処理の詳細な内容について説明する。

#### [実分割数の算出]

15 図9のステップS 1 6と図10のステップS 2 5とに示した実分割数の算出処理の詳細について説明する。

本実施の形態では、立体形状を構成する面の形を表す辺を、パラメータ3次式 $F(t)$  ( $0 \leq t \leq 1$ )で定義する。パラメータ3次式を用いれば、辺の中間点の算出が容易となる。すなわち、パラメータ3次式に $t = 1/2$ を与えることにより、容易に中間点の座標を（CPUに負荷を与えず）求めることができる。

20 ここで、パラメータ3次式で与えられた辺が曲線であれば、その辺は、立体形状を表示する際に複数の線分に分割される（ポリライン（折れ線）近似される）。このとき、辺の分割数が多いほど、分割された線分の形が元の曲線に近くなる（誤差が小さくなる）。ただし、分割数を多くすると、表示ポリゴン数が増大し、

25 処理負荷が大きくなる。そこで、再現される立体形状の精度と処理負荷とのバランスを考えて許容できる誤差（許容誤差（Tol））をコンピュータ100に与え、その許容誤差に応じた適切な分割数（誤差を超えない範囲の最も小さい分割数）を決定する必要がある。

一般に、与えられた許容誤差内で最も小さい分割数を算出するには、負荷の大

きい計算が必要となる。例えば、半径Rの円の分割数nは、

$$n = \pi / \cos^{-1} (1 - \text{Tol} / R) \quad \dots (1)$$

- 5      で求めることができる。この式(1)を、便宜上真の式と呼ぶこととする。ここで、 $\pi$ は円周率である。真の式(1)で用いられているアークコサイン( $\cos^{-1}$ )は、CPU負荷の大きな関数である。その為、本実施の形態では、分割数の計算を高速にする近似式を用いた。その近似式は以下の通りである。

$$n \doteq \pi \times \sqrt{R / 2 \text{Tol}} \quad \dots (2)$$

- 10      ここで、式(2)には、変数として半径Rが含まれている。辺の形は単純な円弧に限らず、自由曲線の場合もある。そのため、式(2)をそのまま用いると、実分割数の決定前に、辺を定義する曲線の曲率(半径R)を算出する必要があるが生じる。そこで、本実施の形態では、予め計算された単位倍率と単位誤差における基底分割数(実数)を用い、半径Rを用いずに分割数nを算出する。

- 15      図12は、空間曲線のポリライン近似の例を示す図である。図12では、空間曲線51を4つの線分52～55に分割している。このときの誤差を  $\text{tol0}$  とする。

- このように、任意の空間曲線51を誤差  $\text{tol0}$  でポリライン(折れ線)近似した場合における分割数n(ポリラインのVector本数)が判っているものとする  
20      (図12の例では分割数  $n = 4$ )。この既知の分割数nを、基底分割数  $n_0$  とする。そして、このときの空間曲線51のサイズを基準サイズと定める。また、基底分割数  $n_0$  を算出した際の誤差を基準誤差  $\text{tol0}$  とする。

- ユーザは、空間曲線51を基準サイズの何倍の大きさで表示させるのかを、表示倍率として指定する。ユーザから任意に指定される表示倍率を  $\text{Scale}$  とすると、  
25      空間曲線51を  $\text{Scale}$  倍した曲線の分割数  $N_1$  は、  
$$N_1 \doteq \sqrt{\text{Scale}} \times n_0$$

$$\dots (3)$$

で示せる。

また、ユーザは、空間曲線51を画面に表示の際の精度を誤差によって指定す

ることができる。基準誤差  $\text{tol0}$  を誤差  $\text{tol1}$  に変更したときの分割数  $N_2$  は、

$$N_2 \doteq \sqrt{(\text{tol0}/\text{tol1})} \times n_0 \quad \dots (4)$$

で示せる。

- 5      従って、この空間曲線 5 1 を、基準サイズ (Scale=1) で誤差を基準誤差 ( $\text{tol0}$ ) 内とするための基底分割数  $n_0$  が既知である場合、この空間曲線 5 1 を Scale 倍し、許容誤差  $\text{Tol}$  ( $=\text{tol1}/\text{tol0}$ ) でポリライン近似する場合の曲線の実際の分割数 (実分割数)  $N$  は、

$$N \doteq \sqrt{(\text{Scale}/\text{Tol})} \times n_0 \quad \dots (5)$$

10

で示せることになる。許容誤差  $\text{Tol}$  は、誤差の大きさを基準誤差  $\text{tol0}$  との比で示している。

ここで、 $n_0$  は表示倍率や許容誤差に依存しない曲線固有の特性値である。そのため、 $n_0$  を基底分割数と称し、予め計算しておくことにより表示倍率や許容  
15      誤差が変更された場合の分割数計算を高速に行うことができる。

15

なお、 $\sqrt{(\text{Scale}/\text{Tol})}$  の値は、3次元空間内の全ての表示対象物に対して1つの値であるため、表示対象物 (たとえば、自動車) を構成する立体形状 (たとえば、部品) の辺 (空間曲線) 毎に計算する必要はない。すなわち、1つの辺に対する表示倍率と許容誤差に応じた実分割数  $N$  を、乗算 1 回で求めることができる。ここで、基底分割数  $n_0$  は正の実数であり、実分割数  $N$  は小数点以下を切り  
20      上げた自然数である。

20

さらに、本実施の形態では、分割パターン処理の簡素化の為に、実分割数は2のべき乗に繰上げる。実分割数を2のべき乗にしておけば、辺の2分割を繰り返すことで、実分割数に応じた辺の等分が可能となる。

- 25      また、直線は基底分割数  $n_0 = 0$  とする。これにより、表示倍率 (Scale) にも許容誤差 (Tol) にも影響されず、実分割数  $N$  は常に 0 となる。これは、分割しないことを意味する。

以下に、図形の分割例について説明する。

図 1 3 は、図形の分割例を示す図である。図 1 3 では、簡単な図形の例として、

円 6 0 を示している。ここで、円 6 0 の半径 R を変えながら、真の式 (1) を用いて分割数 n を求めてみる。得られた分割数を、円 6 0 の半径 R に応じた基底分割数 n 0 とする。このとき、基準誤差 tol0 を 1mm とする。この場合、半径 R の変化と基底分割数 n 0 との関係は、以下の表のようになる。

5

半径 R	1000	100	10	5	2	1	0.5	0.1
誤差 Tol	1	1	1	1	1	1	1	1
基底分割数 n0	72.24	2.20	6.97	4.88	3.00	2.00	1.00	-

この表に示すように、半径 R が大きくなるほど基底分割数 n 0 も大きくなる。なお、半径 R が 0.5 mm 未満の場合アークコサインの計算を行うことができないため、計算不能となる。

10 このようにして得られた基底分割数 n 0 から、表示倍率と許容誤差とに応じた実分割数 N を計算する。

たとえば、半径 10 mm の円を 100 倍 (Scale=100) の大きさに、許容誤差 1 mm (Tol=1) で近似する場合、基底分割数 n 0 = 6.97 を用いて実分割数 N が計算される。各数値を式 (5) の変数に設定して、近似式を解くと、

15 
$$N \doteq \sqrt{100} \times 6.97 = 69.7$$

・・・(6)

となる。実分割数 N の小数点以下を切り上げると 70 になる。ここで、表示倍率を 100 倍にしたときの分割数は、半径が 100 倍の円の分割数と同等になるべきである。この例では、半径 10 mm の 100 倍の半径 1000 mm を有する円  
20 の分割数を真の式 (1) で求めた場合、70.24 である。この値の小数点以下を切り上げると 71 となる。そのため、十分な精度で近似されていることがわかる。

次に、真の式 (1) では計算不能だった半径 0.1 mm の円を 1 mm の誤差で分割する場合の分割数も、半径 10 mm の円を 1 / 100 倍して表示することで  
25 近似できる筈であるから

$$n_0(r=0.1) \div 1/\sqrt{(100)} \times 6.97 = 0.697$$

$$\dots (7)$$

と求めることができる。

[分割パターン決定処理]

- 5 次に、図11のステップS32、S33に示す分割パターンの決定処理について詳細に説明する。

[3辺形分割パターン決定処理]

- まず、3辺形分割パターン決定処理について詳細に説明する。本実施の形態では、全ての辺の分割数が2のべき乗となっているため、全ての3辺形は、4種類の分割パターンの組み合わせによって分割することができる。

図14は、3辺形の分割パターンの種別を示す図である。図14には、3辺の分割数Nの相互関係に応じた分割パターンを、分割図によって示している。

- 第1の3辺形分割パターンは、3辺の分割数がN、 $N \times a_1$ 、 $N \times a_2$ （Nは1より大きい2のべき乗）の関係にある場合である。ここで、 $a_1$ 、 $a_2$ は、2のべき乗である。すなわち、3辺を同じ分割数Nで分割できる場合である。この場合、分割図に示すように、分割数Nに応じてパラメータ3次式で表現された各辺の分割点の座標と分割点の法線ベクトルが求められ、さらに3辺形の間分割線が求められる。そして、中間分割線により元の3辺形が多数の3辺形に分割される。

- 20 第2の3辺形分割パターンは、3辺の分割数が1、N、 $N \times a_3$ （Nは1より大きい2のべき乗）の関係にある場合である。ここで、 $a_3$ は、2のべき乗である。すなわち、分割数が1より大きい2辺を同じ分割数Nで分割できる場合である。この場合、分割図に示すように、パラメータ3次式で表現された分割すべき2辺に関して、分割数Nに応じて分割点の座標と分割点の法線ベクトルが求められ、さらに3辺形の間分割線が求められる。そして、中間分割線により元の3
- 25 辺形が多数の3辺形に分割される。

第3の3辺形分割パターンは、3辺の分割数が1、1、N（Nは1より大きい2のべき乗）の関係にある場合である。すなわち、1辺のみを分割する場合である。この場合、分割図に示すように、パラメータ3次式で表現された分割すべき

1 辺に関して、分割数Nに応じて分割点の座標と分割点の法線ベクトルが求められ、さらに3 辺形の中間分割線が求められる。そして、中間分割線により元の3 辺形が多数の3 辺形に分割される。

第4の3 辺形分割パターンは、3 辺の分割数が全て1 の場合である。この場合、  
5 元の3 辺形は分割されない。

以上のような4 パターン（第4の3 辺形分割パターンでは分割は行われなため、実質3 パターン）の分割を段階的に繰り返すことで、各辺の実分割数に応じて3 辺形が分割される。なお、3 辺形を分割する場合に採用する優先順は、第1  
10 の分割パターン、第2の分割パターン、第3の分割パターンの順番である。

図1 5 は、3 辺形分割パターン決定処理の手順を示すフローチャートである。  
この処理は、図1 1 のステップS 3 2 の処理である。なお、図1 1 では、ステップS 3 5  
15 において、再分割処理が再帰的に実行されることで多段階の分割が行われる（1 段階で1 つの分割パターンによる分割が行われる）が、図1 5 では、分割方法を分かり易く説明するために、多段階の分割処理を一連のフローチャート  
15 で示している。以下、図1 5 に示す処理をステップ番号に沿って説明する。

〔ステップS 5 1〕再分割部1 1 6 は、実分割数の小さい順に、辺を並べ替える（ソートする）。ここで、実分割数の小さい順に、各辺の分割数を  $i$  ,  $j$  ,  $k$  とする（ $i \leq j \leq k$ ）。

〔ステップS 5 2〕再分割部1 1 6 は、 $i$  が1 より大きいかな否かを判断する。  
20  $i$  が1 より大きい場合には、処理がステップS 5 3 に進められる。 $i$  が1 以下の場合には、処理がステップS 5 5 に進められる。

〔ステップS 5 3〕再分割部1 1 6 は、3 辺形の第1の分割パターンによって、各辺を（ $i$  ,  $i$  ,  $i$ ）で分割する。

〔ステップS 5 4〕再分割部1 1 6 は、 $i$  ,  $j$  ,  $k$  の値を更新する。具体的には、 $j / i$  の値を、新たに  $j$  に設定する。 $k / i$  の値を、新たに  $k$  に設定する。  
25  $i$  に1 を設定する。

〔ステップS 5 5〕再分割部1 1 6 は、 $j$  が1 より大きいかな否かを判断する。 $j$  が1 より大きい場合には、処理がステップS 5 6 に進められる。 $j$  が1 以下の場合には、処理がステップS 5 8 に進められる。

[ステップS 5 6] 再分割部 1 1 6 は、3 辺形の第 2 の分割パターンによって、各辺を (1, j, j) で分割する。

[ステップS 5 7] 再分割部 1 1 6 は、j, k の値を更新する。具体的には、 $k/j$  の値を、新たに k に設定する。j に 1 を設定する。

- 5 [ステップS 5 8] 再分割部 1 1 6 は、k が 1 より大きいかなんかを判断する。k が 1 より大きい場合には、処理がステップS 5 9 に進められる。k が 1 以下の場合には、処理が図 1 1 のステップS 3 4 に進められる。

- 10 [ステップS 5 9] 再分割部 1 1 6 は、3 辺形の第 3 の分割パターンによって、各辺を (1, 1, k) で分割する。その後、処理が図 1 1 のステップS 3 4 に進められる。

以上のようにして、3 辺形の面の分割パターンが、各辺の実分割数に応じて決定することができる。分割パターンが決定された後、図 1 1 のステップS 3 4, S 3 5 によって、3 辺形が再分割される。

- 15 たとえば、3 辺形の各辺に関して式 (5) による実分割数を計算したとき、それぞれ、(5, 2 1, 1 2) の結果を得た場合を考える。これらの値は、2 のべき乗に切り上げられ (8, 3 2, 1 6) となる。そして、各辺の分割数を、値の小さい順にソートすると、 $i = 8$ 、 $j = 1 6$ 、 $k = 3 2$  (8, 1 6, 3 2) となる。

- 20 まず、i が 1 より大きいため、第 1 の分割パターンで分割される。すると、各辺が 8 分割される。これにより、i, j, k の値が更新され、(1, 2, 4) となる。

次に、j が 1 より大きいため、第 2 の分割パターンで分割される。すると、j と k とに対応する辺が 2 分割される。これにより、各 3 辺形がさらに複数の 3 辺形に分割される。そして、j, k の値が更新され、(1, 1, 2) となる。

- 25 最後に、k が 1 より大きいため、第 3 のパターンで分割される。すると、k に対応する辺が 2 分割される。これにより、各 3 辺形が複数の 3 辺形に分割される。このようにして、3 辺形を分割することができる。

[4 辺形分割パターン決定処理]

次に、4 辺形の分割パターン決定処理について詳しく説明する。本実施の形態

では、全ての辺の分割数が2のべき乗となっていることから、全ての4辺形は、4種類の分割パターンの組み合わせによって分割することができる。

図16は、4辺形の分割パターンの種別を示す図である。図16には、4辺の分割数Nの相互関係に応じた分割パターンを、分割図によって示している。

- 5 第1の分割パターンは、対向する辺同士を組にしたときに、4辺の分割数が $(M, M \times b_1)$ 、 $(N, N \times b_2)$  ( $M, N$ は1より大きい2のべき乗)の関係にある場合である(括弧内の2つの値が、互いに組となった2つの辺の分割数である)。ここで、 $b_1, b_2$ は、2のべき乗である。すなわち、対向する辺同士を同じ分割数で分割できる場合である。この場合、分割図に示すように、対向する辺の一方の組がMで分割され、他方の組がNで分割される。その結果、元の4
- 10 辺形が複数の4辺形に分割される。

- 第2の分割パターンは、対向する辺同士を組にしたときに、4辺の分割数が $(1, M)$ 、 $(1, N)$  ( $M, N$ は1より大きい2のべき乗)の関係にある場合である。ここで、 $M \leq N$ の関係にあるものとする。すなわち、対向する辺の一方の
- 15 辺を分割する場合である。この場合、分割図に示すように、4辺の内の隣ある2辺が分割対象である。ここで、分割対象の2つの辺が共有する端点から、その端点に対向する端点へ対角線が引かれる。4辺形は、対角線によって2つの3辺形に分割される。対角線の分割数をNとすることで、生成された3つの3辺形を、3辺形の分割パターンに従って分割することができる。具体的には、生成される
- 20 3辺形は、第3のパターンと第4のパターンとに分けられる。

第3の分割パターンは、3辺形の2辺を、それぞれN, Mに分割する場合である。この3辺形は、3辺形の第2の分割パターンと第3の分割パターンとの組み合わせにより、分割することができる。

- 第4の分割パターンは、3辺形の2辺を、それぞれN, Nに分割する場合である。この3辺形は、3辺形の第2の分割パターンによって分割することができる。
- 25

以上のような分割パターンに従って4辺形を分割することができる。

図17は、4辺形分割パターン決定処理の手順を示すフローチャートである。この処理は、図11のステップS33の処理である。なお、図11では、ステップS35において、再分割処理が再帰的に実行されることで多段階の分割が行わ



れる（１段階で１つの分割パターンによる分割が行われる）が、図１７では、分割方法を分かり易く説明するために、多段階の分割処理を一連のフローチャートで示している。以下、図１７に示す処理をステップ番号に沿って説明する。

５      [ステップＳ６１] 再分割部１１６は、対辺を組にする。そして、同じく組となった２つの辺の分割数の比（大きい方の分割数／小さい方の分割数）を、組同士で比較する。そして、分割数の比が小さい方から順に、各組の辺を並べる。同じ組の中の辺同士では、分割数の小さい順に並べる。このとき並べられた各辺の分割数を、配列の順に  $i$ 、 $j$ 、 $k$ 、 $l$  とする。分割数  $i$  の辺と  $j$  の辺とが組を構成し、分割数  $k$  の辺と  $l$  の辺とが組を構成する。このとき、 $i \leq j$ 、 $k \leq l$ 、 $j / i \leq l / k$  の関係となる。

        [ステップＳ６２] 再分割部１１６は、 $i$  と  $k$  との少なくとも一方が１より大きいか否かを判断する。いずれか一方が１より大きければ、処理がステップＳ６３に進められる。両方とも１以下であれば、処理がステップＳ６５に進められる。

１５      [ステップＳ６３] 再分割部１１６は、第１の４辺形分割パターンにより、４辺形を分割する。すなわち、分割数  $i$ 、 $j$  との辺は  $i$  で分割され、分割数  $k$ 、 $l$  との辺は  $k$  で分割される。

        [ステップＳ６４] 再分割部１１６は、 $i$ 、 $j$ 、 $k$ 、 $l$  の値を更新する。具体的には、 $j / i$  の値を新たに  $j$  に設定する。 $l / k$  の値を新たに  $l$  に設定する。 $i$  に１を設定する。 $k$  に１を設定する。

２０      [ステップＳ６５] 再分割部１１６は、第２の４辺形分割パターンに基づいて、ステップＳ６３による分割で生成された４辺形を２分割する。これにより、各４辺形から２つの３辺形が生成される。

        [ステップＳ６６] 再分割部１１６は、ステップＳ６５で生成された３辺形を１つ選択する。

２５      [ステップＳ６７] 再分割部１１６は、ステップＳ６６で選択した３辺形に対して、３辺形分割パターン決定処理を行う。たとえば、選択した３辺形が、４辺形に関する第３の分割パターンに従っていれば、その３辺形は、３辺形の第２の分割パターンと第３の分割パターンによって分割パターンが決定される。また、選択した３辺形が、４辺形に関する第４の分割パターンに従っていれば、その３

辺形は、3 辺形の第 2 の分割パターンによって分割パターンが決定される。

5    [ステップ S 6 8] 再分割部 1 1 6 は、ステップ S 6 5 で生成された全ての 3 辺形を分割したか否かを判断する。全ての 3 辺形の分割パターン決定処理が終了した場合には、処理が図 1 1 のステップ S 3 4 に進められる。未処理の 3 辺形が  
あれば、処理がステップ S 6 6 に進められる。

以上のようにして、4 辺形の面の分割パターンを、各辺の実分割数に応じて決定することができる。分割パターンが決定された後、図 1 1 のステップ S 3 4、S 3 5 によって、4 辺形が再分割される。

たとえば、3 辺形の各辺に関して式 (5) による実分割数を計算したとき、そ  
10    れぞれ、(6, 2, 1 2, 7) の結果を得た場合を考える。ここで、分割数 6 の辺と分割数 2 の辺とが対向する辺であり、分割数 1 2 の辺と分割数 7 の辺とが対向する辺である。これらの値は、2 のべき乗に切り上げられ (8, 2, 1 6, 8) となる。そして、分割数の比が小さい方の組を先にして、各組内での各辺を、分割数の小さい順にソートすると、 $i = 8$ ,  $j = 1 6$ ,  $k = 2$ ,  $l = 8$  (8, 1  
15    6, 2, 8) となる。

まず、 $i$  と  $k$  が 1 より大きいため、4 辺形の第 1 の分割パターンで分割される。すると、分割数  $i$  の辺と  $j$  の辺とが 8 分割される。同様に、分割数  $k$  の辺と  $j$  の辺とが 2 分割される。これにより、 $i$ ,  $j$ ,  $k$ ,  $l$  の値が更新され、(1, 2, 1, 4) となる。

20    次に、第 2 の分割パターンによって、第 1 の分割パターンによる分割で生成された各 4 辺形が、対角線で 2 分割される。対角線に対しては、1 の値と同じ分割数が設定される。これにより、(1,  $j$ , 1) の分割数の 3 辺形と、(1, 1, 1) の分割パターンの 3 辺形とに分けられる。すなわち、(1, 2, 4) の分割パターンの 3 辺形と (1, 4, 4) の分割パターンの辺形が生成される。

25    分割数 (1, 2, 4) の 3 辺形に関しては、 $j$  ( $= 2$ ) が 1 より大きいため、3 辺形の第 2 の分割パターンで分割される。すると、 $j$  と 1 とに対応する辺が 2 分割される。これにより、3 辺形がさらに複数の 3 辺形に分割される。そして、 $j$ , 1 の値が更新され、(1, 1, 2) となる。そして、1 が 1 より大きいため、3 辺形の第 3 のパターンで分割される。すると、1 に対応する辺が 2 分割される。

分割数 (1, 4, 4) の 3 辺形に関しては、1 (= 4) が 1 より大きいため、3 辺形の第 2 の分割パターンで分割される。すると、1 に対応する辺が 4 分割される。これにより、3 辺形がさらに複数の 3 辺形に分割される。

以上のようにして、4 辺形が複数の 3 辺形に分割される。

#### 5 [パラメータ 3 次式決定法]

辺を分割する場合、その辺の端点に対して設定されている法線ベクトルに応じてパラメータ 3 次式が決定される。そして、辺の曲線を定義したパラメータ 3 次式におけるパラメータ  $t$  ( $0 \leq t \leq 1$ ) に、0.5 を設定することで、その辺の中点を求める。なお、パラメータ 3 次式の決定法は、立体形状における面同士の隣接関係に応じて異なる。

・第 1 の隣接関係は、隣接する面同士が滑らかに接続している場合である。この場合、法線ベクトルから接線ベクトルを経由して求める方法と、法線ベクトルから 2 次微分ベクトルを仮定して求める方法がある。

・第 2 の隣接関係は、隣接する面同士が共有する辺で折れている場合である。

15 ・第 3 の隣接関係は、辺の形状を強制的に指示する必要がある場合である。

以下、面同士の隣接関係に応じた、パラメータ 3 次式の決定法について説明する。

[隣接する面同士が滑らかに接続している場合]

図 18 は、隣接する面同士が滑らかに接続している立体形状の一例を示す図である。図に示すように、立体形状 71 は、互いに隣接する面  $F1a$ 、 $F1b$  を有している。隣接する面  $F1a$ 、 $F1b$  が辺  $S1(t)$  を共有し、さらに辺  $S1(t)$  の両端点における法線ベクトル  $N10$ 、 $N11$  も隣接する面  $F1a$ 、 $F1b$  同士が共有する。この場合、この 2 つの隣接する面  $F1a$ 、 $F1b$  は、辺  $S1(t)$  において滑らかに接続されていると判断される。

25 このような場合、以下の 2 つの方法の何れかでパラメータ 3 次式を決定することができる。

まず、法線ベクトルから接線ベクトルを経由してパラメータ 3 次式を求める方法について説明する。

この方法では、両端点の座標  $P10$ 、 $P11$ 、両端点における法線ベクトル  $N$

10, N11、及び接線の長さ  $r_{10}$ ,  $r_{11}$  を与え、接線ベクトル  $T_{10}$ ,  $T_{11}$  を決定する。なお、端点座標、法線ベクトル、および接線の長さは、立体構造データ記憶部 120 内の端点情報において定義されている。

5 この場合、接線ベクトル  $T_{10}$  は、法線ベクトル  $N_{10}$  に垂直で、座標  $P_{10}$  から座標  $P_{11}$  の方向に最も近い方向を向いた、長さ  $r_{10}$  のベクトルである。また、接線ベクトル  $T_{11}$  は、法線ベクトル  $N_{11}$  に垂直で、座標  $P_{10}$  から座標  $P_{11}$  の方向に最も近い方向を向いた、長さ  $r_{11}$  のベクトルである。

式で示すと以下の通りとなる。

$$10 \quad T_{10} = r_{10} \cdot N_{10} \times (P_{11} - P_{10}) \times N_{10} / |N_{10} \times (P_{11} - P_{10}) \times N_{10}|$$

$$\dots (8)$$

$$T_{11} = r_{11} \cdot N_{11} \times (P_{11} - P_{10}) \times N_{11} / |N_{11} \times (P_{11} - P_{10}) \times N_{11}|$$

$$\dots (9)$$

15 式 (8)、式 (9) において、“ $\times$ ” はベクトルの外積を示しており、“ $\cdot$ ” はスカラー値 ( $r_0$  や  $r_1$ ) の乗算を示している。

次に、両端点の座標 ( $P_{10}$ ,  $P_{11}$ ) 及び両端点における接線ベクトル ( $T_{10}$ ,  $T_{11}$ ) を用いて、以下の4つのベクトル式を満たしたパラメータ3次式  $S_1(t)$  を決定する。

$$20 \quad S_1(t=0) = P_{10}$$

$$S_1(t=1) = P_{11}$$

$$dS_1/dt(t=0) = T_{10}$$

$$dS_1/dt(t=1) = T_{11}$$

$$\dots (10)$$

25 このように、端点での条件を決定することで、3次元区間内の曲線を決定することができる。たとえば、ファーガソン曲線を算出することができる。

次に、法線ベクトルから2次微分ベクトルを仮定してパラメータ3次式を求める方法について説明する。

図19は、2次微分ベクトルを用いて決定されたパラメータ3次式の一例を示

す図である。図 19 に示すように、両端点における 2 次微分ベクトル  $W_{20}$ ,  $W_{21}$  が、両端点における法線ベクトル  $N_{20}$ ,  $N_{21}$  と方向が近似している（正負反対の場合も含む）と仮定（弧長媒介変数による 2 次微分ではないので一致しない）することができる。このとき、法線ベクトル  $N_{20}$ ,  $N_{21}$  にベクトルの  
 5 大きさ  $r_{20}$ ,  $r_{21}$  を乗じたベクトルを 2 次微分ベクトル  $W_{20}$ ,  $W_{21}$  とする。

なお、法線ベクトル  $N_{20}$ ,  $N_{21}$  は常に面の表を向いているため、ベクトルの大きさ  $r_{20}$ ,  $r_{21}$  を与える時に、必要に応じて負の値とすることで 2 次微分方向に合わせる。たとえば、曲線が、立体形状の表方向に凸であれば、ベクトルの大きさ  $r_{20}$ ,  $r_{21}$  を負の値とする。逆に、曲線が、立体形状の表方向に  
 10 凹であれば、ベクトルの大きさ  $r_{20}$ ,  $r_{21}$  を正の値とする。なお、ベクトルの大きさ  $r_{20}$ ,  $r_{21}$  は、図 6 に示す端点情報 426 の接線定義情報 426c 内に予め設定される。

2 次微分ベクトル  $W_{20}$ ,  $W_{21}$  は、それぞれ法線ベクトル  $N_{20}$ ,  $N_{21}$  と  
 15 平行で、長さと向きが  $r_{20}$ ,  $r_{21}$  で示されるベクトルと近似する。これを式で示すと以下の通りとなる。

$$W_{20} = r_{20} \cdot N_{20} \quad \dots (11)$$

$$W_{21} = r_{21} \cdot N_{21} \quad \dots (12)$$

両端点の座標  $P_{20}$ ,  $P_{21}$ 、及び両端点における 2 次微分ベクトル  $W_{20}$ ,  $W_{21}$  を用いて、以下の 4 つのベクトル式からパラメータ 3 次式  $S_2(t)$  を決定する。

$$\begin{aligned} S_2(t=0) &= P_{20} \\ S_2(t=1) &= P_{21} \\ d^2 S_2 / dt^2 (t=0) &= W_{20} \\ d^2 S_2 / dt^2 (t=1) &= W_{21} \\ &\dots (13) \end{aligned}$$

このような条件を満たしたパラメータ 3 次式  $S_2(t)$  が決定される。

[隣接する面同士が共有する辺で折れている場合]

次に、隣接する面同士が共有する辺で折れている場合のパラメータ 3 次式の決定方法について説明する。

図 20 は、隣接する面同士が共有する辺で折れている立体形状の一例を示す図である。図 20 に示す立体形状 72 は、面 F 3 a と面 F 3 b とが辺 S 3 (t) で隣接している。面 F 3 a の法線ベクトル N 3 0 a, N 3 1 a には、辺 S 3 (t) を湾曲させる成分を持っていない。一方で隣接関係にある面 F 3 b の法線ベクトル N 3 0 b, N 3 1 b は辺 S 3 (t) を湾曲させる成分を持っている。

このように隣接する面 F 3 a, F 3 b によって共有される辺 S 3 (t) の両端点の法線ベクトルが、2つの面 F 3 a, F 3 b によって共有されない場合、この隣接する面 F 3 a, F 3 b は共有される辺 S 3 (t) で折れていると判断される。この場合、4本の法線ベクトルを用いて接線ベクトルの方向を決定する。

接線ベクトル T 3 0 は法線ベクトル N 3 0 a, N 3 0 b の外積の方向で、長さは r 3 0 のベクトルである。同様に、接線ベクトル T 3 1 は法線ベクトル N 3 1 a, N 3 1 b の外積の方向で、長さは r 3 1 のベクトルである。なお、長さ r 3 0 は、端点 3 0 に定義された接線の長さであり、長さ r 3 1 は、端点 3 1 に定義された接線の長さである。これを式で示すと以下の通りとなる。

$$T 3 0 = r 3 0 \cdot N 3 0 a \times N 3 0 b / |N 3 0 a \times N 3 0 b|$$

・・・(14)

$$T 3 1 = r 3 1 \cdot N 3 1 a \times N 3 1 b / |N 3 1 a \times N 3 1 b|$$

・・・(15)

次に、両端点の座標 (P 3 0, P 3 1)、及び両端点における接線ベクトル (T 3 0, T 3 1) を用いて、以下の4つのベクトルを満たしたパラメータ 3 次式 S 3 (t) を決定する。

$$\begin{aligned} S 3 (t=0) &= P 3 0 \\ S 3 (t=1) &= P 3 1 \\ d S 3 / d t (t=0) &= T 3 0 \\ d S 3 / d t (t=1) &= T 3 1 \\ &・・・(16) \end{aligned}$$

[辺の形状を強制的に指示する必要がある場合]

次に、辺の形状を強制的に指示する必要がある場合について説明する。

図 2 1 は、辺の形状を強制的に指示された立体形状の一例を示す図である。図 2 1 に示す立体形状 7 3 は、面 F 4 a と面 F 4 b とが隣接している。隣接する面 F 4 a, F 4 b が辺 S 4 (t) を共有し、さらに辺 S 4 (t) の両端点における法線ベクトル N 4 0, N 4 1 も隣接する面 F 4 a, F 4 b によって共有される。この場合、この 2 つの隣接する面 F 4 a, F 4 b は辺 S 4 (t) において滑らかに接続されていると判断されるが、法線ベクトル N 4 0, N 4 1 には辺 S 4 (t) を湾曲させる成分を持っていない。

10    そういった場合においても辺 S 4 (t) を強制的に湾曲させる必要があれば接線ベクトル T 4 0, T 4 1 を直接与える。

両端点の座標 P 4 0, P 4 1、及び両端点における接線ベクトル T 4 0, T 4 1 を用いて、以下の式を満たすパラメータ 3 次式 S 4 (t) を決定する。

$$\begin{aligned} S_4(t=0) &= P_{40} \\ 15 \quad S_4(t=1) &= P_{41} \\ dS_4/dt(t=0) &= T_{40} \\ dS_4/dt(t=1) &= T_{41} \\ &\dots (17) \end{aligned}$$

20    以上のようにして、立体形状を構成する辺のパラメータ 3 次式を決定することができる。

[辺上の任意の中間座標と法線ベクトルの求め方]

25    辺の曲線を定義したパラメータ 3 次式が決定されると、先に決定された分割パターンに従って、辺が分割される。分割された辺の端点には、新たに端点の位置と法線ベクトルとを定義する必要がある。以下、中間点における座標（中間座標）と法線ベクトルとの算出方法について説明する。

[中間座標の求め方]

中間座標の求め方を、図 2 2 を参照して説明する。

図 2 2 は、分割対象の面の一例を示す図である。図 2 2 の例では、分割される面 8 1 は 3 辺形である。面 8 1 は、端点 P 5 0, P 5 1 を有する辺 8 1 a、端点

P 5 1, P 5 2 を有する辺 8 1 b、端点 P 5 2, P 5 0 を有する辺 8 1 c で構成されている。端点 P 5 0 での法線ベクトルが V 5 0、接線ベクトルが N 5 0 である。端点 P 5 1 での法線ベクトルが V 5 1、接線ベクトルが N 5 1 である。

- 5 パラメータ 3 次式の決定法に従い端点 P 5 0 から端点 P 5 1 を結ぶ辺 8 1 a のパラメータ 3 次式 S 5 (t) が決定されている場合、辺 8 1 a 上の任意の点である P c は、パラメータ 3 次式 S 5 (t) の t 値 (中点なら t = 0.5) を定めることにより決定される。

次に、中間点の法線ベクトルの求め方について、図 2 2 を参照して説明する。

- 10 任意の中間点における法線ベクトルの決定法としては、以下の 4 通りの決定法が考えられる。

第 1 の方法は、中間点の法線ベクトル N c が N 5 0 から N 5 1 に線形変化していると想定する方法である。その計算式は、以下の通りである。

$$N c = (1 - t) N 5 0 + (t) N 5 1$$

$$\dots (18)$$

- 15 第 2 の方法は、中間点の法線ベクトル N c が N 5 0 から N 5 1 に線形変化し、中間点の接線ベクトル V c と直交していると想定する方法である。その計算式は、以下の通りである。

$$N c = V c \times (N 5 0 + N 5 1) \times V c$$

$$\dots (19)$$

- 20 第 3 の方法は、中間点の法線ベクトル N c が、N 5 0 方向から N 5 1 方向に回転していると想定する方法である。その計算式は、以下の通りである。

$$N c = V c \times N 5 0 \times N 5 1$$

$$\dots (20)$$

但し、N 5 0 × N 5 1 = 0 の場合の計算式は、以下の通りである。

25  $N c = V c \times N 5 0 \times V c$

$$\dots (21)$$

第 4 の方法は、中間点の法線ベクトル N c が、N 0 方向から N 1 方向に 3 次で変化していると想定する方法である。その計算式は、以下の通りである。

$$N c = -2 (N 5 1 - N 5 0) t^3 + 3 (N 5 1 - N 5 0) t^2 + N 5 0$$



• • • (2 2)

式(18)～(22)で得られたベクトル $N_c$ を単位ベクトルに正規化したものが、中間点の法線ベクトルとなる。

### 「3 辺形の任意中間曲線の計算方法」

5      3 辺形を分割すると、元の 3 辺形内に新たな辺が生成される。そのとき、生成された辺の曲線式を計算する必要がある。

図 2 3 は、分割される 3 辺形を含む立体形状の一例を示す図である。図 2 3 の例では、円柱の立体形状 8 2 が示されている。立体形状 8 2 の側面が、複数の 3 辺形で構成される。側面に含まれる 1 つの 3 辺形 8 2 a は、端点 P 6 0、P 6 1、P 6 2 を有している。ハイブリッドメッシュデータにおいて定義されている 3 辺形 8 2 a は、たとえば、画像で表示するときに複数の 3 辺形に分割される。

ここで、辺  $8\ 2\ b\ a$  上の中間点の座標  $P\ 7\ 0$  と法線ベクトル  $N\ 7\ 0$ 、および辺  $8\ 2\ b\ b$  上の中間点の座標  $P\ 7\ 1$  と法線ベクトル  $N\ 7\ 1$  を用いて、2つの辺  $8\ 2\ b\ a$ 、 $8\ 2\ b\ b$  の中間点同士を結ぶ辺  $8\ 2\ c\ a$  のパラメータ 3 次式を求めることができる。この際、パラメータ 3 次式の決定法としては、たとえば、隣接する面同士が滑らかに接続している場合の決定法により求める。

但し、座標P70での接線の長さ $r_{70}$ と、座標P71での接線の長さ $r_{71}$ とは、以下の式で求める。

$$r_{70} = r_{60} \times |P_{71} - P_{70}| / |P_{61} - P_{60}| \quad \dots (23)$$

$$r_{71} = r_{61} \times |P_{71} - P_{70}| / |P_{61} - P_{60}| \quad \dots (24)$$

なお、 $r_{60}$ は、端点 $P_{60}$ での接線の長さであり、 $r_{61}$ は端点 $P_{61}$ での接線の長さである。

25 [3 辺形の任意中間曲線の計算方法 (他の案)]

他の 3 辺形の任意中間曲線の計算方法もある。

図 2 4 は、分割される 3 辺形の一例を示す図である。図 2 4 の三辺形 8 3 について、P 8 0、P 8 1、P 8 2 は、三辺形の頂点（3 辺形を構成する辺の端点）を示す。N 8 0、N 8 1、N 8 2 は、P 8 0、P 8 1、P 8 2 における法線ベク

トルを示す。T 8 1, T 8 2は、P 8 1とP 8 2とを端点とする辺の各端点における接線ベクトルを示す。U 8 1, U 8 0は、P 8 1とP 8 0とを端点とする辺の各端点における接線ベクトルを示す。V 8 2, V 8 0は、P 8 2とP 8 0とを端点とする辺の各端点における接線ベクトルを示す。

- 5      ここで、内部で生成される情報として、e 8 0, e 8 1, e 8 2がある。e 8 0, e 8 1, e 8 2は、三角形を形成する辺を示し、パラメータ 3 次式で示される。

ここで、内部中間曲線 e 8 3 を、以下の方法で求める。

- 10    [ S T 1 ] e 8 2 上を、P 8 0 から P 8 1 に向かって、 $u$  ( $0 < u < 1$ ) だけ進んだ点 P 8 3 を求める。

[ S T 2 ] P 8 3 における e 8 2 の接線ベクトル U 8 3 を求める。

[ S T 3 ] P 8 3 における法線ベクトル N 8 3 を求める。

[ S T 4 ] P 8 3 において U 8 3, N 8 3, T 8 3 の方向関係が、P 8 1 における U 8 1, N 8 1, T 8 1 の関係と等しくなる接線ベクトル T 8 3 を求める。

- 15    [ S T 5 ] 接線ベクトル T 8 3 を求めた方法と同様に、P 8 2 における V 8 2, N 8 2, T 8 2 の関係と等しくなる接線ベクトル T 8 4 を求める。

[ S T 6 ] 接線ベクトル T 8 3, T 8 4 から、e 8 3 のパラメータ 3 次式を求める。

- 20    このようにして、3 辺形を分割するための境界となる辺のパラメータ 3 次式を求めることができる。

[ 4 辺形の内部中間座標と法線ベクトルの求め方 ]

次に、4 辺形の内部中間座標と法線ベクトルの求め方について説明する。

まず、図 2 5 を参照して 4 辺形の任意中間座標の求め方について説明する。

- 25    図 2 5 は、分割される 4 辺形の一例を示す図である。ここで、4 辺形 8 4 を表す式を、 $S_9(u, v)$  とする。4 辺形 8 4 が  $Coon's$  曲面とすると、図 2 5 のように各端点における座標 P 9 0 ~ P 9 3 と  $u$  方向ベクトル U 9 0 ~ U 9 3、 $v$  方向ベクトル V 9 0 ~ V 9 3 を用いて以下のベクトル式で示せる。

$$S_9(u=0, v=0) = P_{90}$$

$$S_9(u=0, v=1) = P_{91}$$

$$S_9(u=1, v=0) = P_{92}$$

$$S_9(u=1, v=1) = P_{93}$$

... (25)

$$\partial S_9 / \partial u (u=0, v=0) = U_{90}$$

$$5 \quad \partial S_9 / \partial u (u=0, v=1) = U_{91}$$

$$\partial S_9 / \partial u (u=1, v=0) = U_{92}$$

$$\partial S_9 / \partial u (u=1, v=1) = U_{93}$$

... (26)

$$\partial S_9 / \partial v (u=0, v=0) = V_{90}$$

$$10 \quad \partial S_9 / \partial v (u=0, v=1) = V_{91}$$

$$\partial S_9 / \partial v (u=1, v=0) = V_{92}$$

$$\partial S_9 / \partial v (u=1, v=1) = V_{93}$$

... (27)

これらの式を連立させて解くことにより、 $S_9(u, v)$  が求まり、 $(u=u_0, v=v_0)$  で示される任意点の座標  $P$  は、 $P = S_9(u=u_0, v=v_0)$  で与えられる。

次に、4 辺形の任意中間点における法線ベクトルの決定法について説明する。  
 $(u=u_0, v=v_0)$  で示される任意点  $P$  における法線ベクトルは、 $u$  方向接線ベクトル  $\partial S_9 / \partial u (u=u_0, v=v_0)$  と、 $v$  方向接線ベクトル  $\partial S_9 / \partial v (u=u_0, v=v_0)$  との外積で与えられる。

以上のような処理を行うことで、ハイブリッドメッシュデータに基づいた立体形状の画面表示が可能となる。

[画面表示例]

以下、図 26～図 30 を参照して、立体形状の画面表示例について説明する。  
 本実施の形態では、立体形状を構成する辺のみを画面表示することもできる。辺のみを画面表示することで、立体形状の構造を画面上で視覚的に容易に認識することができる。

例として、球体の立体形状を 3 次元 CAD で作成した場合について説明する。  
 球体を示す CAD データは、CAD データ変換部 114 によって、ハイブリッド

メッシュデータに変換される。球体をハイブリッドメッシュデータに変換すると、たとえば、8つの3辺形からなる8面体で表現される。

図26は、ハイブリッドメッシュデータで定義されている立体形状の辺を表示した画面例を示す図である。図26の画面には、8面体の立体形状をハイブリッドメッシュデータで定義したときの立体形状の辺が表示されている（ワイヤフレームだけを表示）。このような構造を定義したハイブリッドメッシュデータを任意の精度で再分割して、立体形状を画面に表示することができる。

たとえば、立体形状を画面内で回転させている場合や、表示倍率が表示に小さい場合には、ハイブリッドメッシュデータの再分割を行わずに立体形状が表示される。

図27は、ハイブリッドメッシュデータで定義されている立体形状を表示した画面例を示す図である。この画面には、図26に示した構造のハイブリッドメッシュデータをそのまま（再分割を行わずに）表示したときの立体形状が表示されている。図27の例では、再分割を行っていないため、8つの面で立体形状が表現されている。これにより、立体形状を高速に画面表示することができる。

一方、表示倍率を上げて、静止した立体形状を画面表示する場合には、ハイブリッドメッシュデータの再分割が行われる。再分割によって、球体を表す立体形状が大量の面で再現される。

図28は、ハイブリッドメッシュデータを再分割したときの立体形状の辺を表示した画面例を示す図である。図28に示す画面には、ハイブリッドメッシュデータの再分割が行われることで、多数の面で構成された球体が表示される（ワイヤフレームだけを表示）。

図28に示すような構成に分割されたデータにも基づいて立体形状を表示すれば、3次元CADで作成したときの球体と同様の見栄えの立体形状を表示することができる。

図29は、ハイブリッドメッシュデータを再分割したときの立体形状を表示した画面例を示す図である。図29に示すように、画面には、再分割を行うことにより、ほとんど球体に見える立体形状が表示される。

ハイブリッドメッシュデータデータは、元のCADデータに比べてデータ容量

が少ないため、元の立体形状を詳細に再現する場合であっても、HDD 103などの記憶容量が少なくてもすむ。また、再分割によってポリゴンデータが生成される毎に3次元表示用API 140に渡されるため、再分割後のポリゴンデータをRAM 102に格納する必要がなく、RAM 102の使用量を低減することができる。すなわち、少ない記憶容量のRAM 102であっても、CADで作成された立体形状を表示させることができる。

ところで、立体形状を画面表示するには、各面の端点に定義されている法線ベクトルを用いて、グーローシェーディングを行うことができる。グーローシェーディングを行うことで、立体形状を構成する1つの面が平面ポリゴンで表示されていても、各面の境界を滑らかに表現することができる。なお、本実施の形態では、法線ベクトルとして単位ベクトルを用いているため、一般的なグーローシェーディングにそのまま利用することができる。

また、立体形状を表示するために必要な資源や処理負荷が少なくてもすむことにより、非常に複雑な立体形状であっても、容易に画面表示することができる。たとえば、ネットワークを介したコラボレーションシステム上でCADを用いた自動車の設計を行い、自動車の構造を表す立体形状を、各共同作業者がネットワークを介して参照することも可能である。

図30は、自動車の立体形状の表示画面例を示す図である。図30に示す画面には、自動車のエンジンルームの構造が表示されている。このように複雑な構造の立体形状であっても、各コンピュータで容易に表示させることができる。

以上のようにして、複雑な立体形状を少ない処理負荷で画面表示することができる。

なお、図10に示したように、本実施の形態ではフェース毎に視点と立体形状との相対位置の変化の有無を判断しているが、立体形状（オブジェクト）毎に相対位置の変化の有無を判断してもよい。その場合、ステップS23の処理に代えて、たとえば、ステップS14とステップS15との間に、視点と立体形状の基準位置との相対位置の変化の有無を判断する処理が行われる。相対位置が変化していれば、処理をステップS18に進め、相対位置が変化していなければ処理をステップS15に進める。

また、仮想 3 次元空間内で視点だけが移動可能な場合などには、視点と立体形状との相対位置の変化の有無を判断する代わりに、視点の移動の有無を判断してもよい。その場合、ステップ S 2 3 の処理に代えて、たとえば、ステップ S 1 3 とステップ S 1 4 との間に視点の移動の有無を判断する処理が行われる。視点が移動していれば、全てのオブジェクトに関して生データ表示処理（ステップ S 1 8）を行い、処理を終了する。視点が移動していなければ、処理をステップ S 1 4 に進める。

このように、表示画面中でオブジェクトが動くか否かの判断処理を単純化することで、処理の高速化を図ることができる。

10     なお、上記の処理機能は、コンピュータによって実現することができる。その場合、コンピュータが有すべき機能の処理内容を記述したプログラムが提供される。そのプログラムをコンピュータで実行することにより、上記処理機能がコンピュータ上で実現される。処理内容を記述したプログラムは、コンピュータで読み取り可能な記録媒体に記録しておくことができる。コンピュータで読み取り可能な記録媒体としては、磁気記録装置、光ディスク、光磁気記録媒体、半導体メモリなどがある。磁気記録装置には、ハードディスク装置（HDD）、フレキシブルディスク（FD）、磁気テープなどがある。光ディスクには、DVD（Digital Versatile Disc）、DVD-RAM(Random Access Memory)、CD-ROM (Compact Disc Read Only Memory)、CD-R (Recordable) / RW (ReWritable) などがある。光磁気記録媒体には、MO(Magneto-Optical disc) などがある。

25     プログラムを流通させる場合には、たとえば、そのプログラムが記録されたDVD、CD-ROMなどの可搬型記録媒体が販売される。また、プログラムをサーバコンピュータの記憶装置に格納しておき、ネットワークを介して、サーバコンピュータから他のコンピュータにそのプログラムを転送することもできる。

プログラムを実行するコンピュータは、たとえば、可搬型記録媒体に記録されたプログラムもしくはサーバコンピュータから転送されたプログラムを、自己の記憶装置に格納する。そして、コンピュータは、自己の記憶装置からプログラムを読み取り、プログラムに従った処理を実行する。なお、コンピュータは、可搬

型記録媒体から直接プログラムを読み取り、そのプログラムに従った処理を実行することもできる。また、コンピュータは、サーバコンピュータからプログラムが転送される毎に、逐次、受け取ったプログラムに従った処理を実行することもできる。

- 5     以上説明したように本発明では、立体形状データに予め定義された面の頂点の位置と頂点における法線方向とに基づいて曲面形状を決定し、その曲面形状に沿った複数の面に分割して立体形状を表示するようにしたため、少ないデータ量の立体形状データから高精度の立体形状を表示させることができる。

- 10    上記については単に本発明の原理を示すものである。さらに、多数の変形、変更が当業者にとって可能であり、本発明は上記に示し、説明した正確な構成および応用例に限定されるものではなく、対応するすべての変形例および均等物は、添付の請求項およびその均等物による本発明の範囲とみなされる。

## 請 求 の 範 囲

1. 物の形状を立体的に表示させる立体形状表示プログラムにおいて、  
コンピュータに、
  - 5 少なくとも1つの面で立体形状を定義した立体形状データに基づいて、前記立体形状を構成する前記面の分割パターンを決定すると共に、前記立体形状データに予め定義された前記面の頂点の位置と当該頂点における法線方向とに基づいて前記面の曲面形状を決定し、  
決定された前記曲面形状を、決定された前記分割パターンによって分割し、前  
10 記曲面形状に沿った複数の面を生成し、  
生成された前記複数の面で構成される立体形状を表示する、  
処理を実行させることを特徴とする立体形状表示プログラム。
  2. 前記分割パターンの決定の際には、前記面を構成する辺毎の分割数を決定し、前記辺毎の分割数に応じて前記面の分割パターンを決定することを特徴とする請求の範囲第1項記載の立体形状表示プログラム。
  - 15 3. 前記分割パターン決定の際には、前記面を構成する前記辺の分割数を、表示精度に関連する操作入力に応じて決定することを特徴とする請求の範囲第2項記載の立体形状表示プログラム。
  4. 前記表示精度に関連する操作入力は、表示倍率の指定であることを特徴とする請求の範囲第3項記載の立体形状表示プログラム。
  - 20 5. 前記表示精度に関連する操作入力は、許容誤差の指定であることを特徴とする請求の範囲第3項記載の立体形状表示プログラム。
  6. 前記表示精度に関連する操作入力は、表示画面内での立体形状の動作指示であることを特徴とする請求の範囲第3項記載の立体形状表示プログラム。
  - 25 7. 前記分割パターン決定の際には、前記面を構成する前記辺に予め設定されている基底分割数を、前記表示精度に関連する操作入力に応じて増減させることで、前記辺の分割数を決定することを特徴とする請求の範囲第3項記載の立体形状表示プログラム。
  8. 前記分割パターン決定の際には、操作入力によって指定された表示倍率と



許容誤差とに応じた係数を決定し、当該係数を前記基底分割数に乗算した値によって前記辺の分割数を決定することを特徴とする請求の範囲第7項記載の立体形状表示プログラム。

9. 前記分割パターン決定の際には、2のべき乗の値の中から前記辺の分割数を決定することを特徴とする請求の範囲第2項記載の立体形状表示プログラム。

10. 前記曲面形状の決定の際には、前記面の頂点に対して予め設定されている接線ベクトルに応じて、前記面を構成する辺の曲線を決定することを特徴とする請求の範囲第1項記載の立体形状表示プログラム。

11. 前記立体形状の表示の際には、表示画面内での当該立体形状の移動の有無を判断し、移動している場合には前記立体形状データに予め定義されている立体形状を表示することを特徴とする請求の範囲第1項記載の立体形状表示プログラム。

12. 物の形状をコンピュータで立体的に表示させるための立体形状表示方法において、

- 15 少なくとも1つの面で立体形状を定義した立体形状データに基づいて、前記立体形状を構成する前記面の分割パターンを決定すると共に、前記立体形状データに予め定義された前記面の頂点の位置と当該頂点における法線方向とに基づいて前記面の曲面形状を決定し、

- 20 決定された前記曲面形状を、決定された前記分割パターンによって分割し、前記曲面形状に沿った複数の面を生成し、

生成された前記複数の面で構成される立体形状を表示する、  
ことを特徴とする立体形状表示方法。

13. 物の形状を立体的に表示する立体形状表示装置において、

- 25 少なくとも1つの面で立体形状を定義した立体形状データに基づいて、前記立体形状を構成する前記面の分割パターンを決定する分割パターン決定手段と、

前記立体形状データに予め定義された前記面の頂点の位置と当該頂点における法線方向とに基づいて、前記面の曲面形状を決定する曲面形状決定手段と、

前記曲面形状決定手段で決定された前記曲面形状を、前記分割パターン決定手段で決定された前記分割パターンによって分割し、前記曲面形状に沿った複数の

面を生成する面生成手段と、

前記面生成手段で生成された前記複数の面で構成される立体形状を表示する表示手段と、

を有することを特徴とする立体形状表示装置。

- 5    1 4.    物の形状を立体的に表示させるための立体形状表示プログラムを記録したコンピュータ読み取り可能な記録媒体において、
- 前記コンピュータに、
- 少なくとも 1 つの面で立体形状を定義した立体形状データに基づいて、前記立体形状を構成する前記面の分割パターンを決定すると共に、前記立体形状データ
- 10    に予め定義された前記面の頂点の位置と当該頂点における法線方向とに基づいて前記面の曲面形状を決定し、
- 決定された前記曲面形状を、決定された前記分割パターンによって分割し、前記曲面形状に沿った複数の面を生成し、
- 生成された前記複数の面で構成される立体形状を表示する、
- 15    処理を実行させることを特徴とする立体形状表示プログラムを記録したコンピュータ読み取り可能な記録媒体。

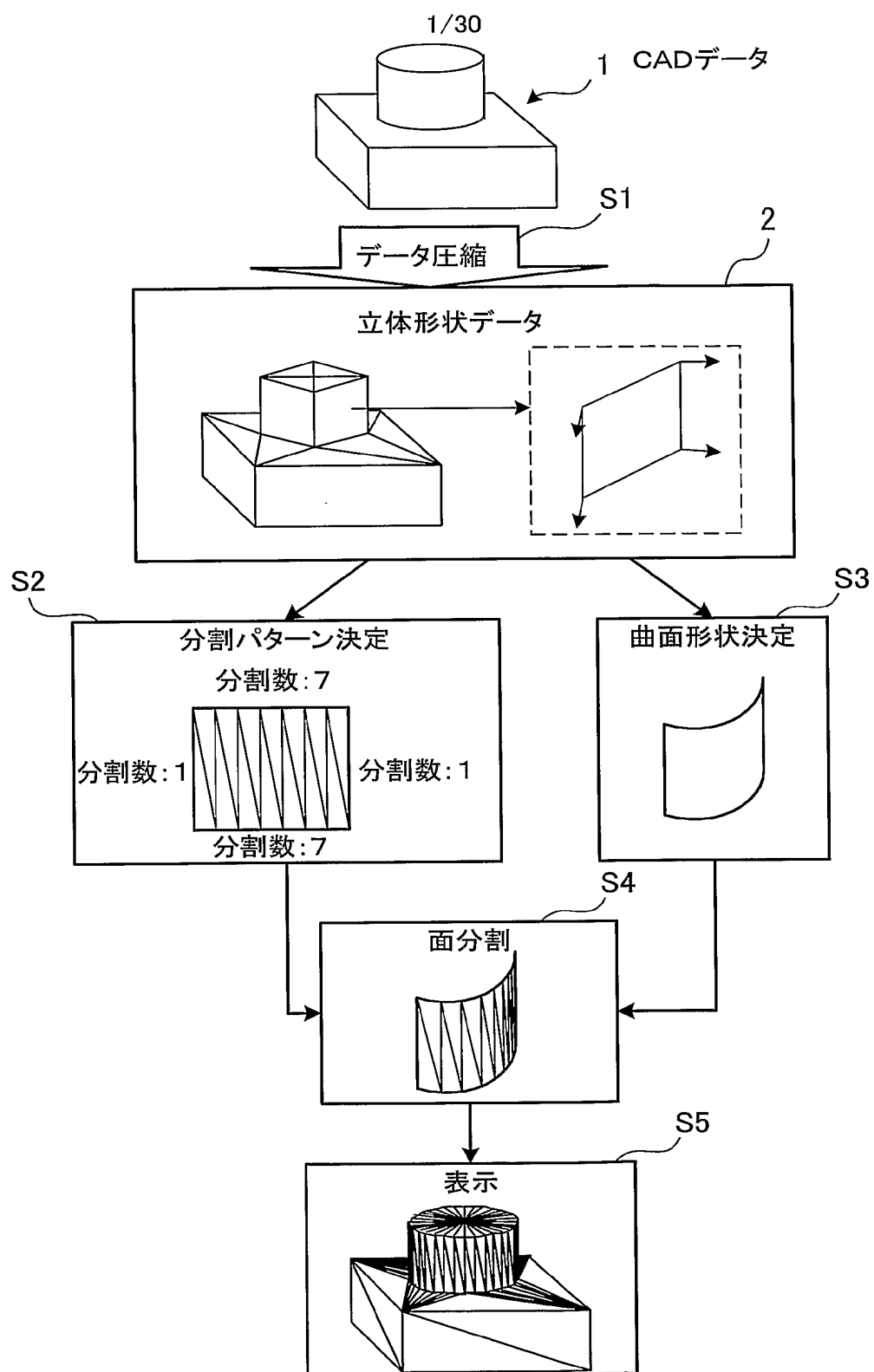


図1

2/30

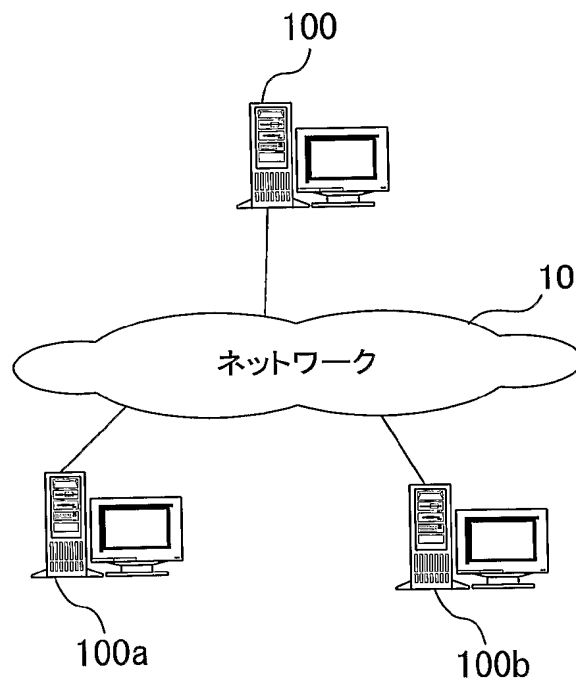


図2

3/30

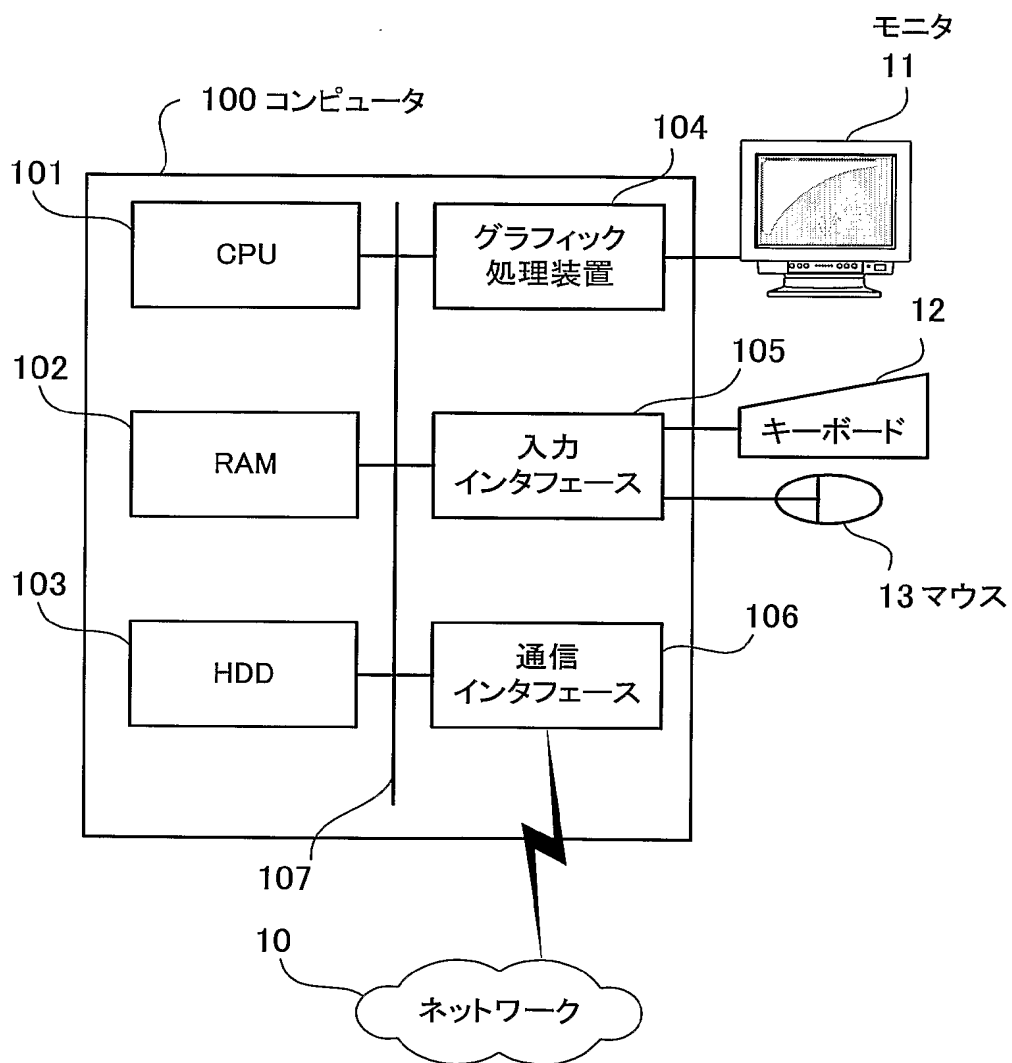


図3

4/30

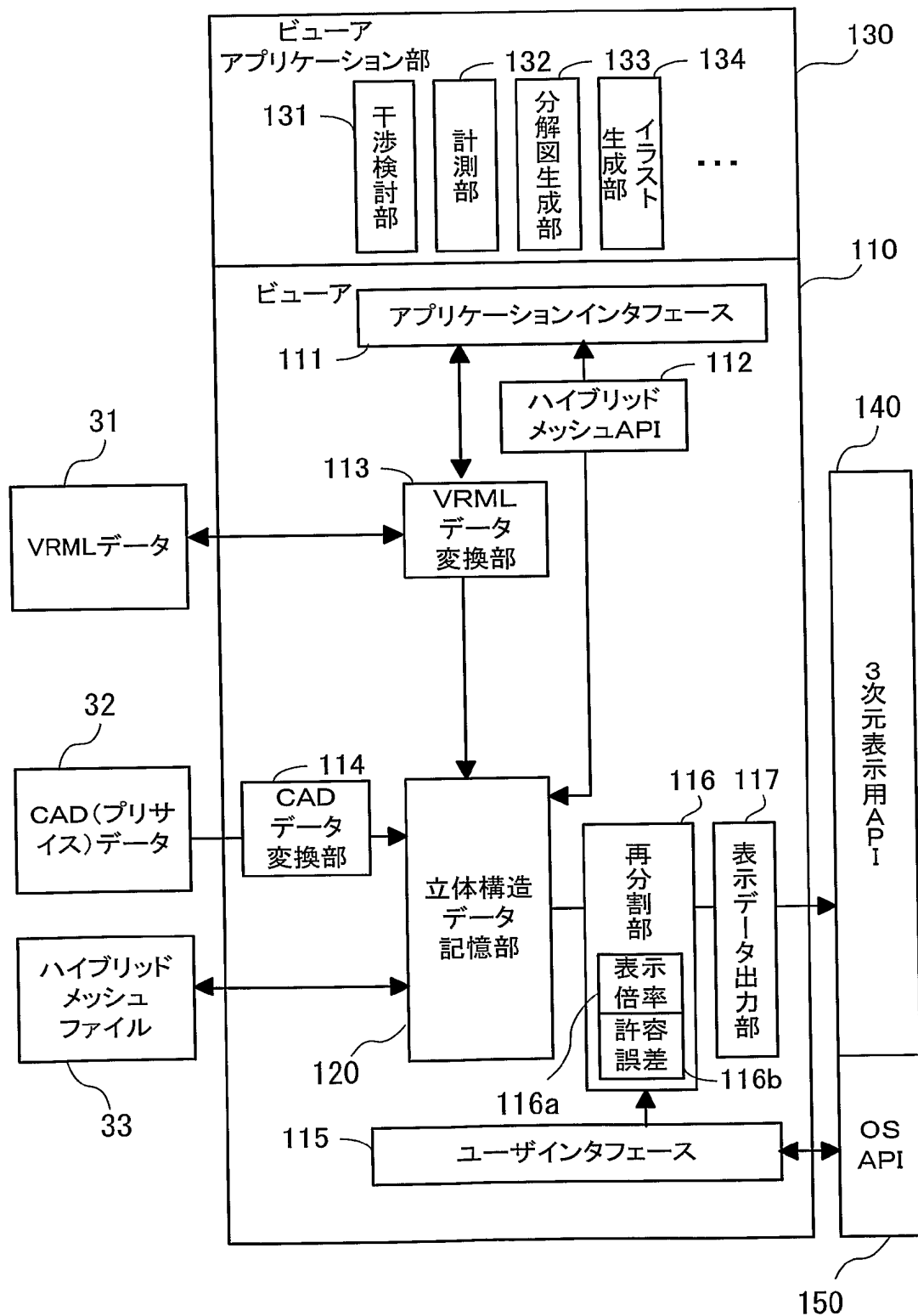


図4

5/30

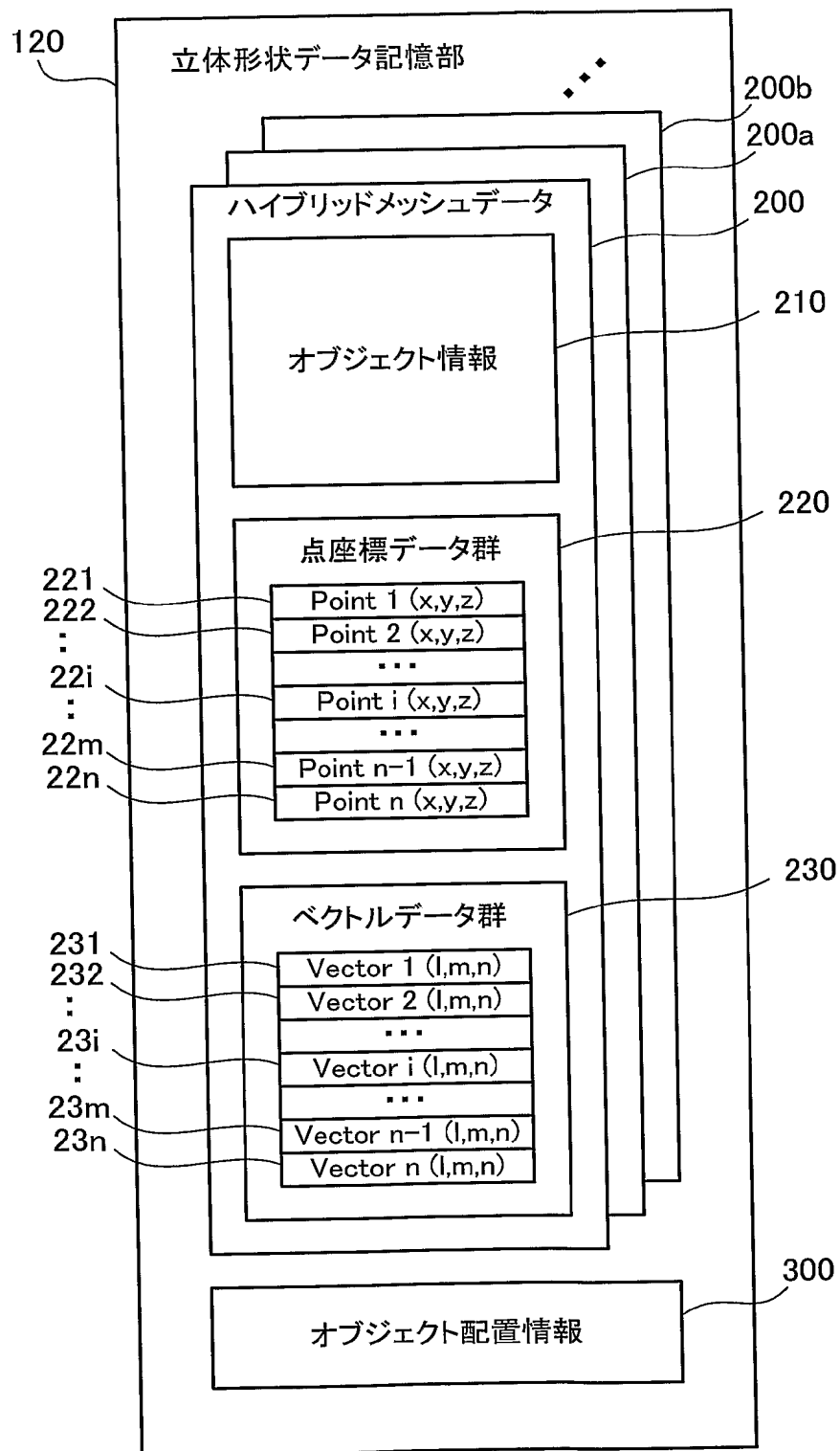


図5

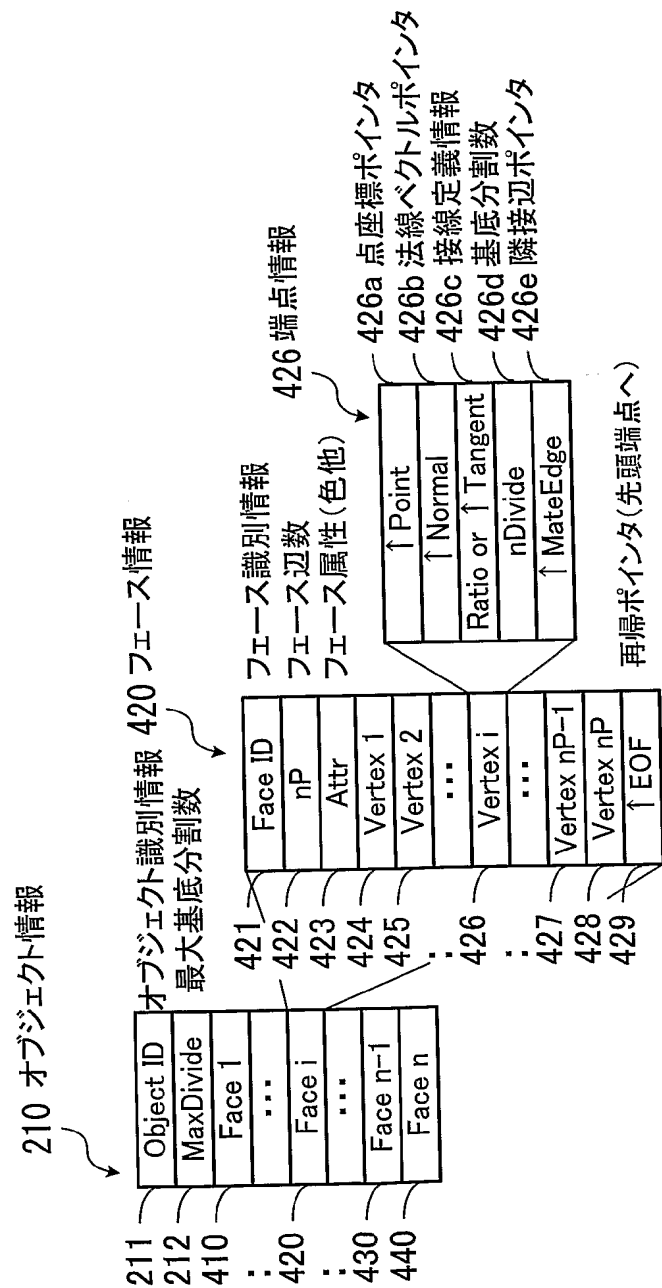
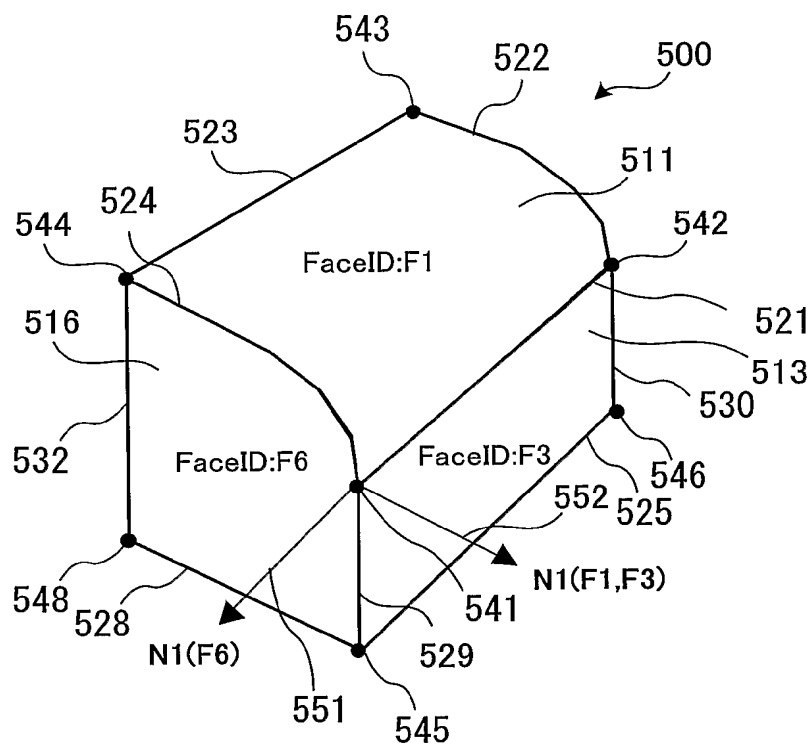


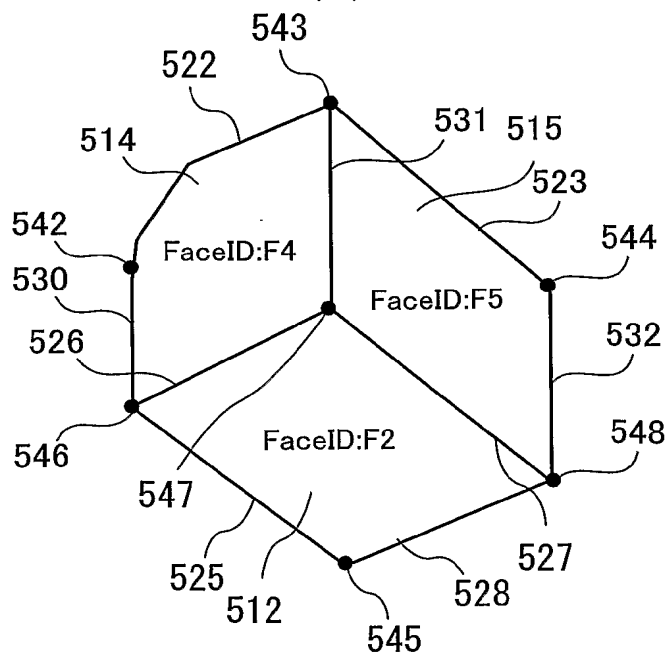
図6



7/30



(A)



(B)

図7

8/30

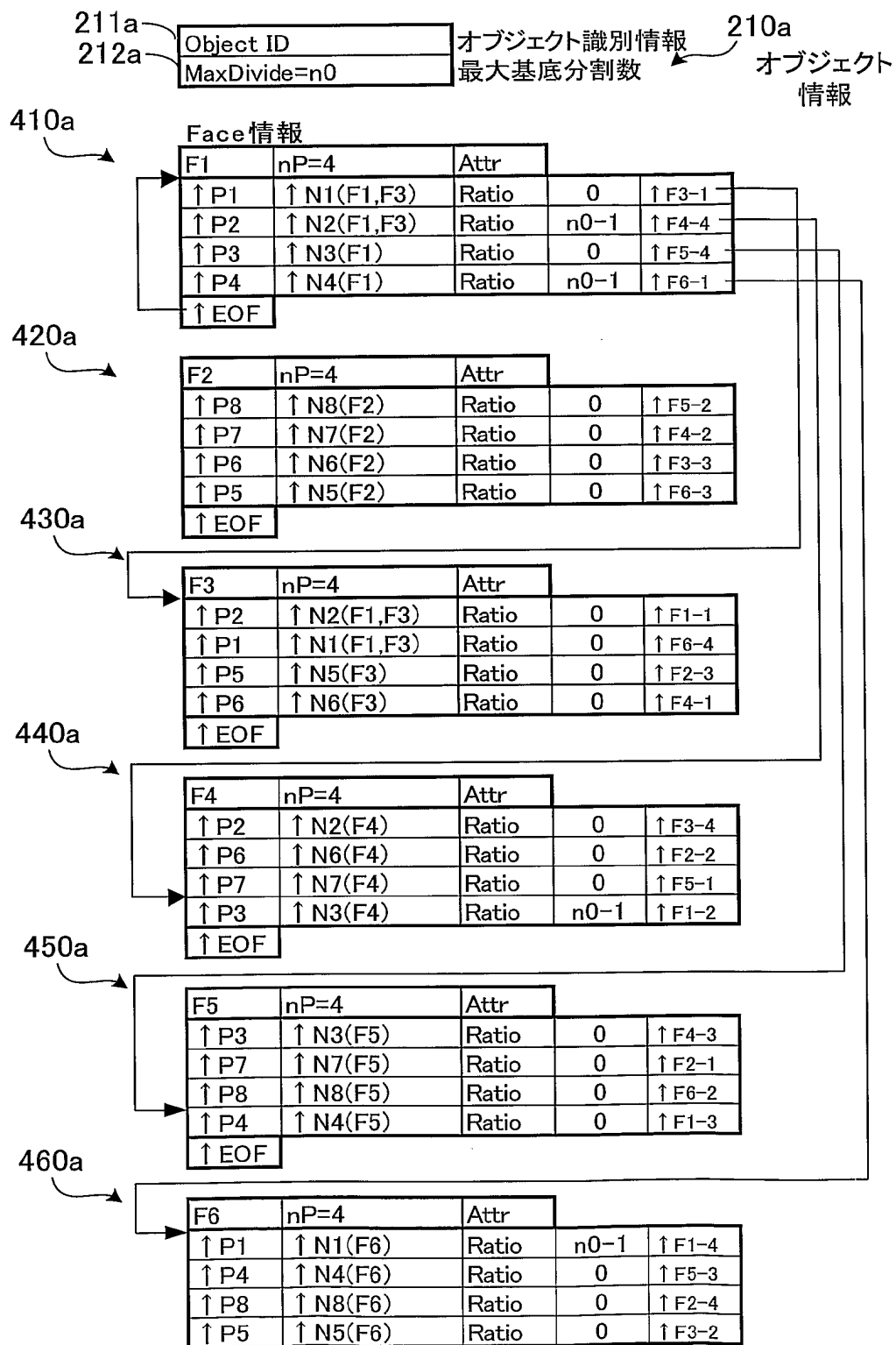


図 8

9/30

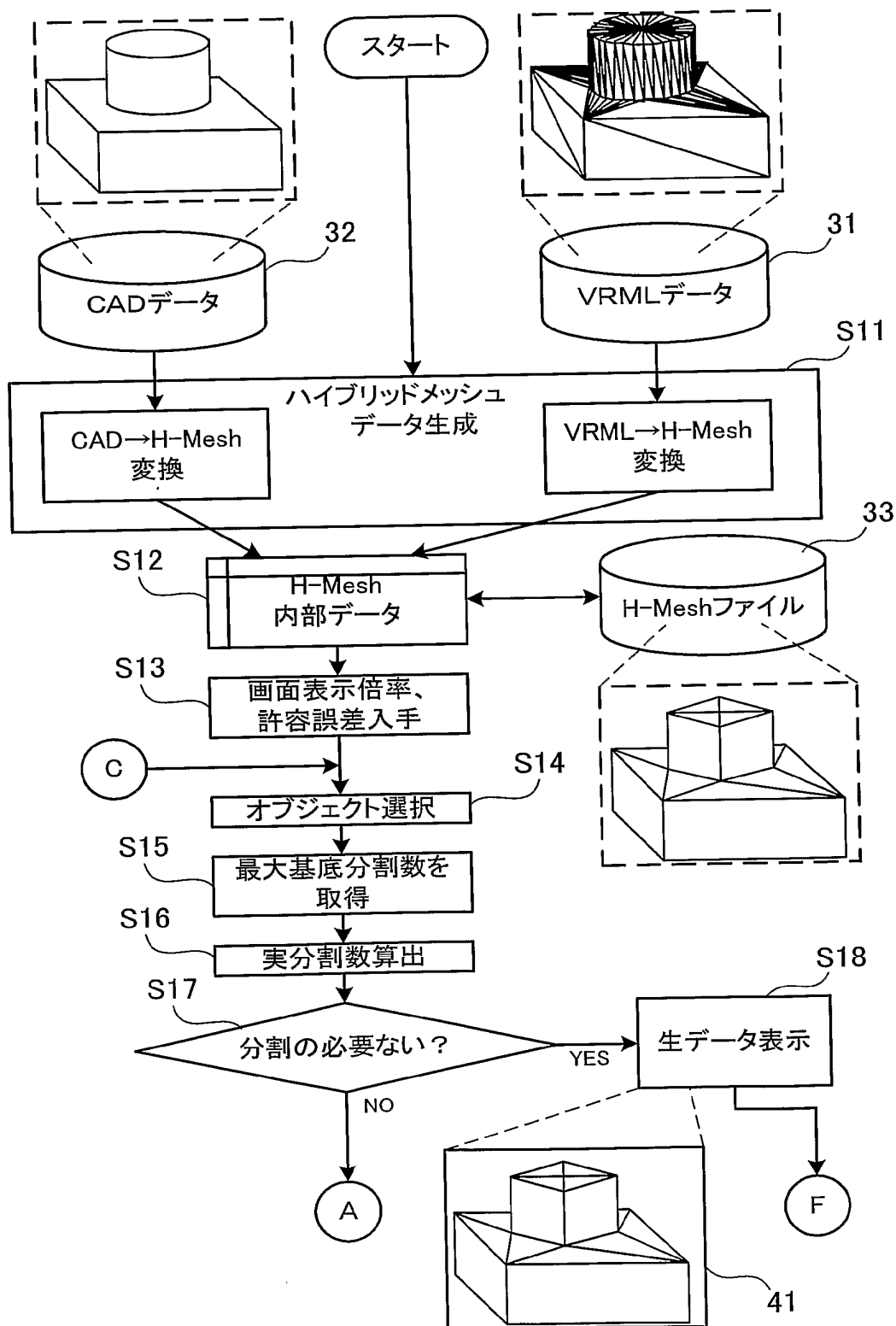


図9

10/30

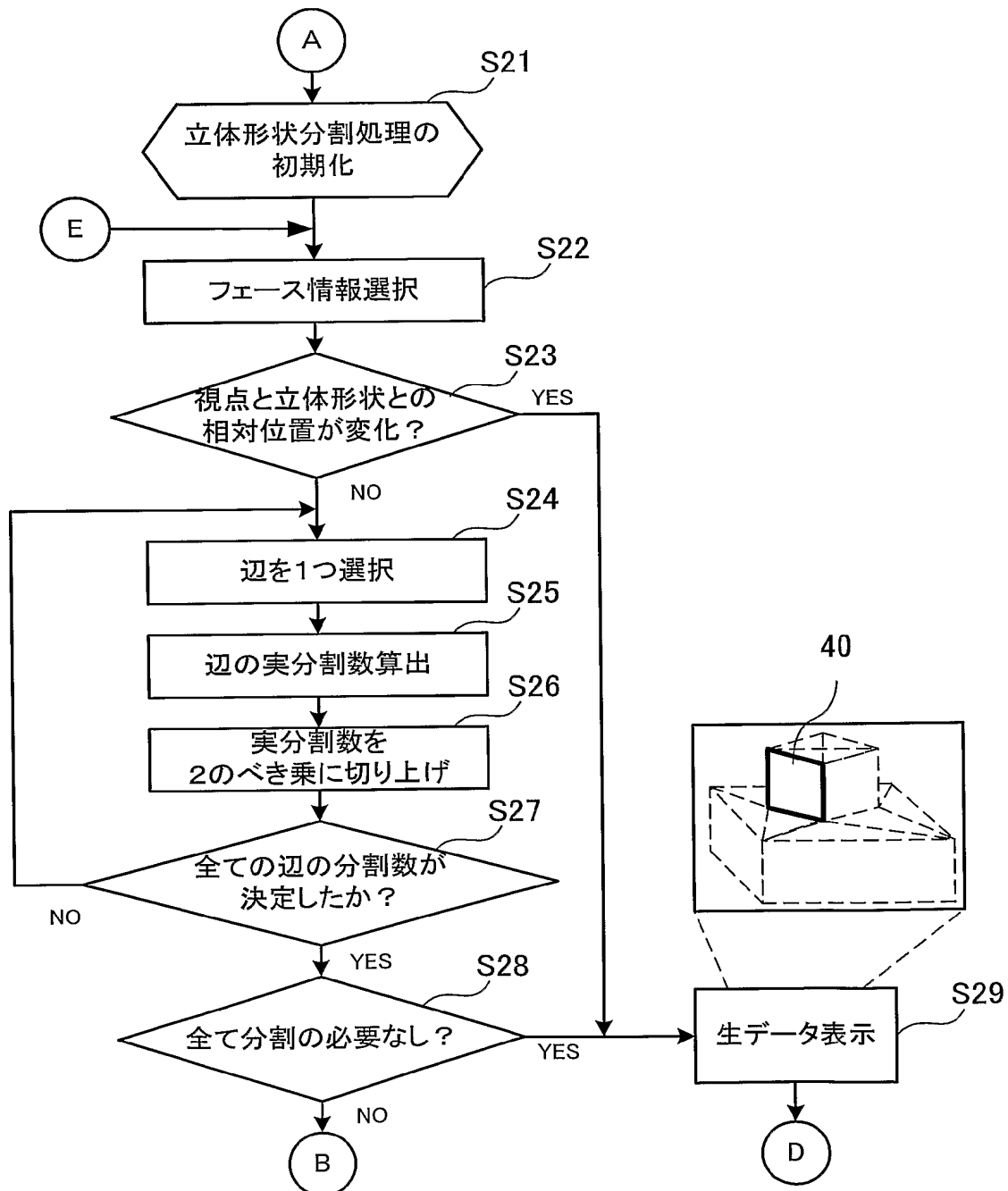


図10

11/30

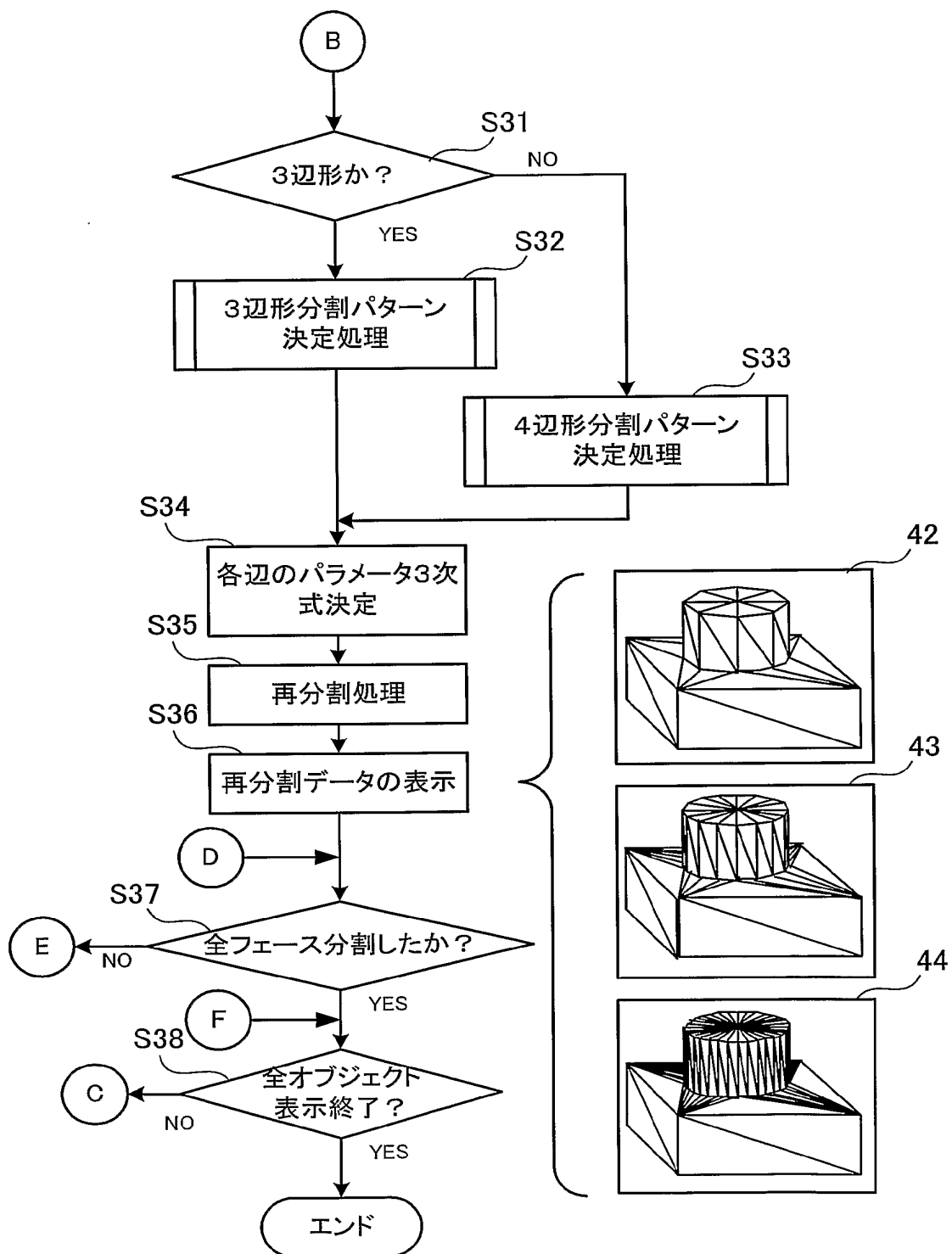


図 11

12/30

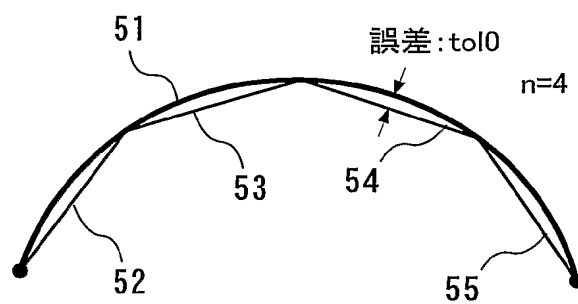


図12

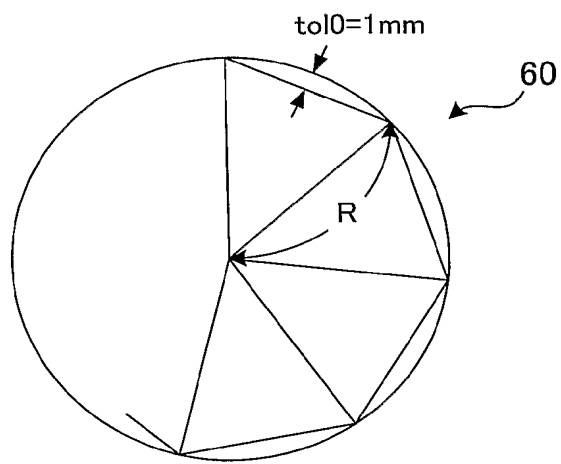


図 13

3 边形分割パターン

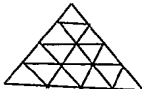
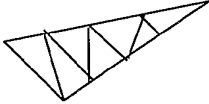
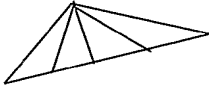
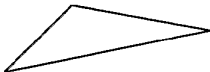
各辺残り分割数	分割パターン	分割図
①N, N×a1, N×a2	N N N	
②1, N, N×a3	1 N N	
③1, 1, N	1 1 N	
④1,1,1	1 1 1	

図 14



15/30

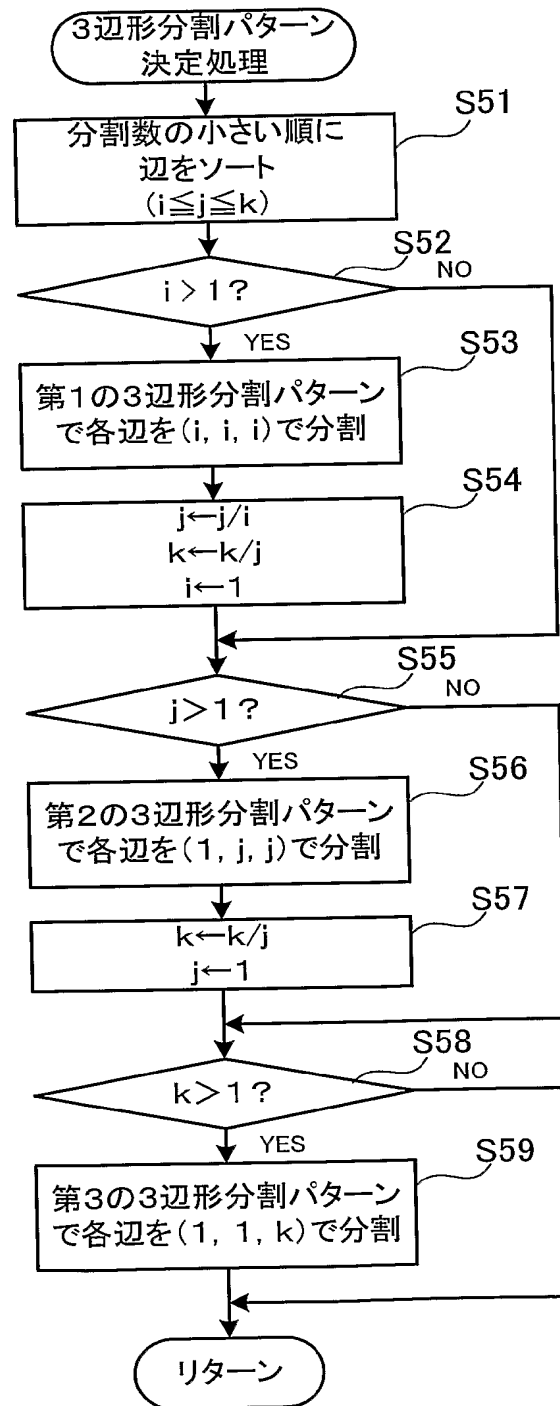


図15

4边形分割パターン

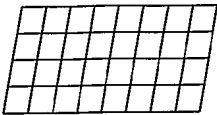
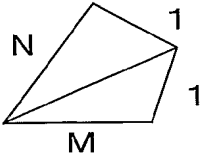
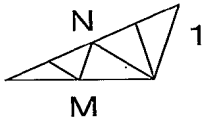

各辺残り分割数	分割パターン	分割図
① M, M×b1, N, N×b2	M M N N	
② 1, M, 1, N	1 M 1 N M ≤ N	
③ 1, M, N	▲ 1 M N	
④ 1, N, N	▲ 1 N N	

図16

17/30

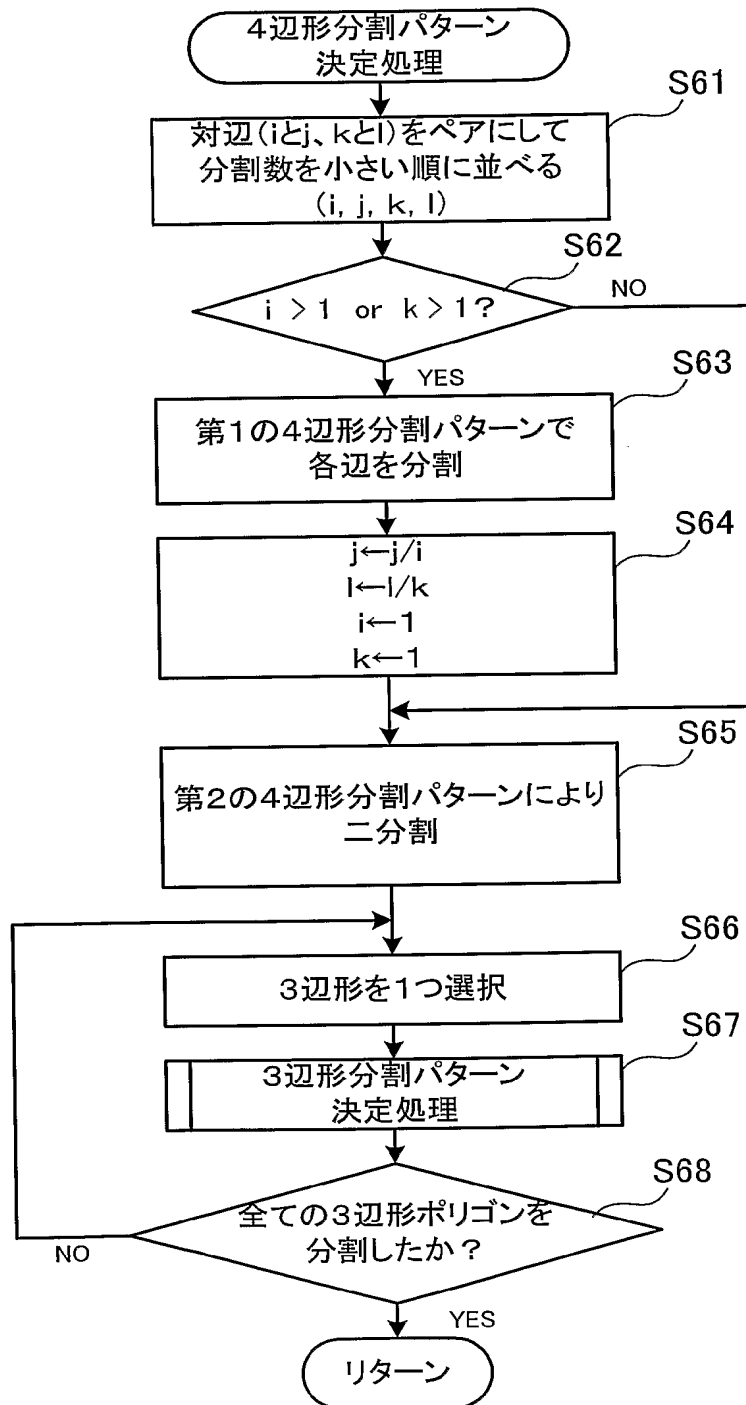


図17

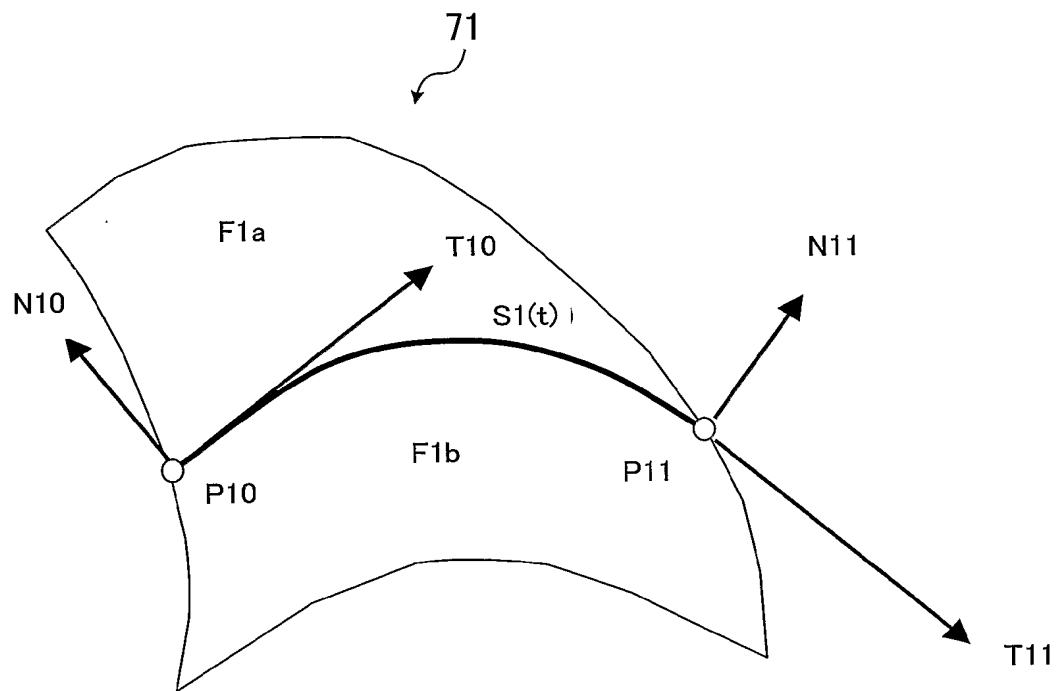


図18

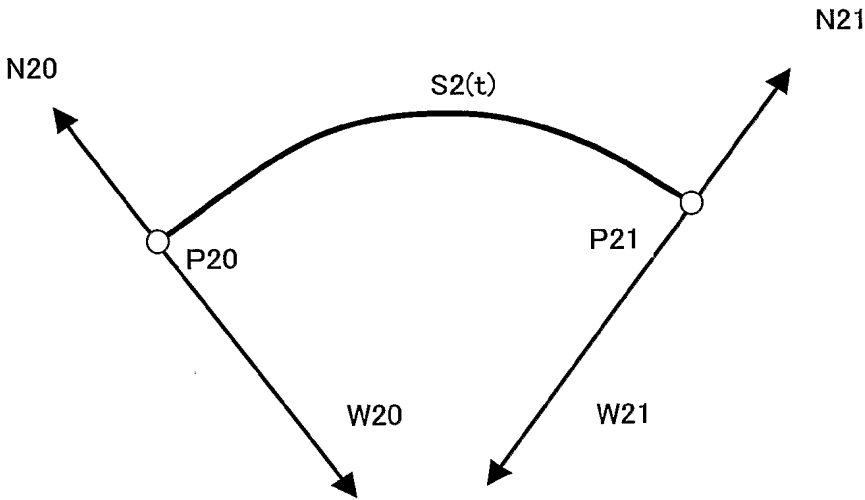


図19

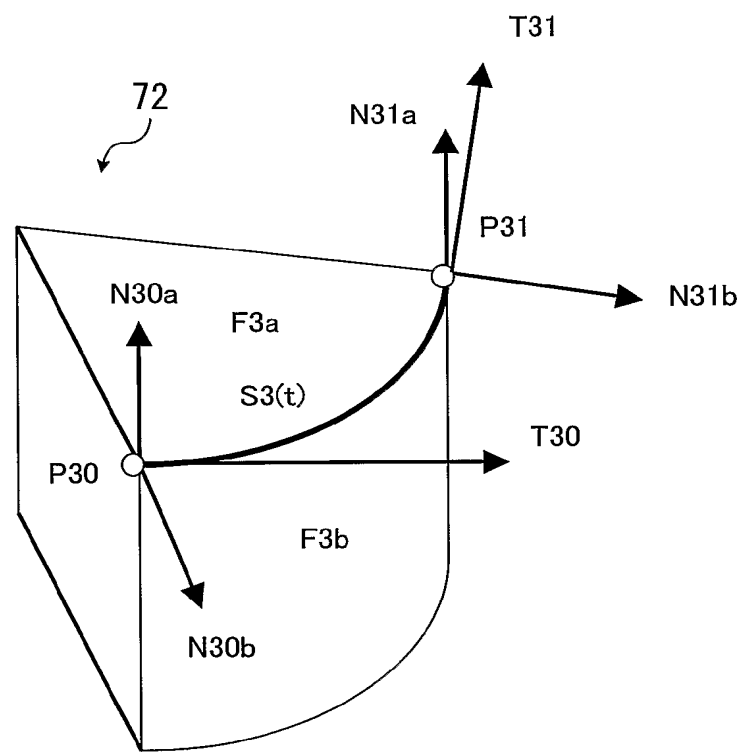


図 20

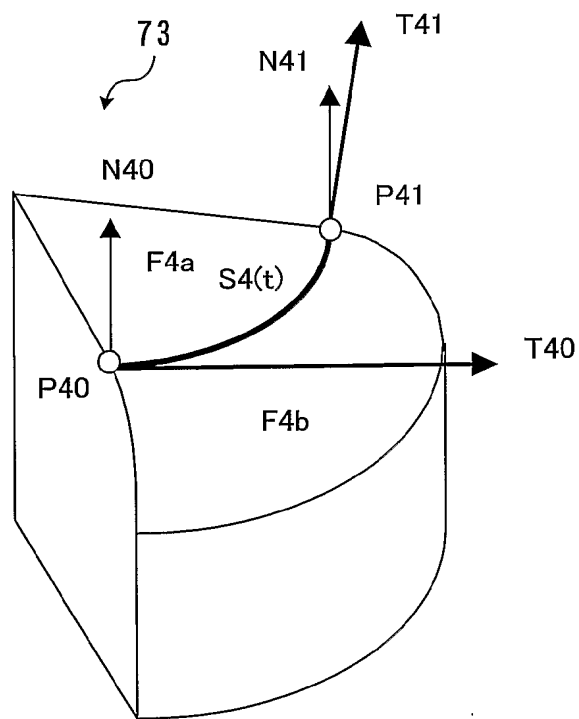


図21

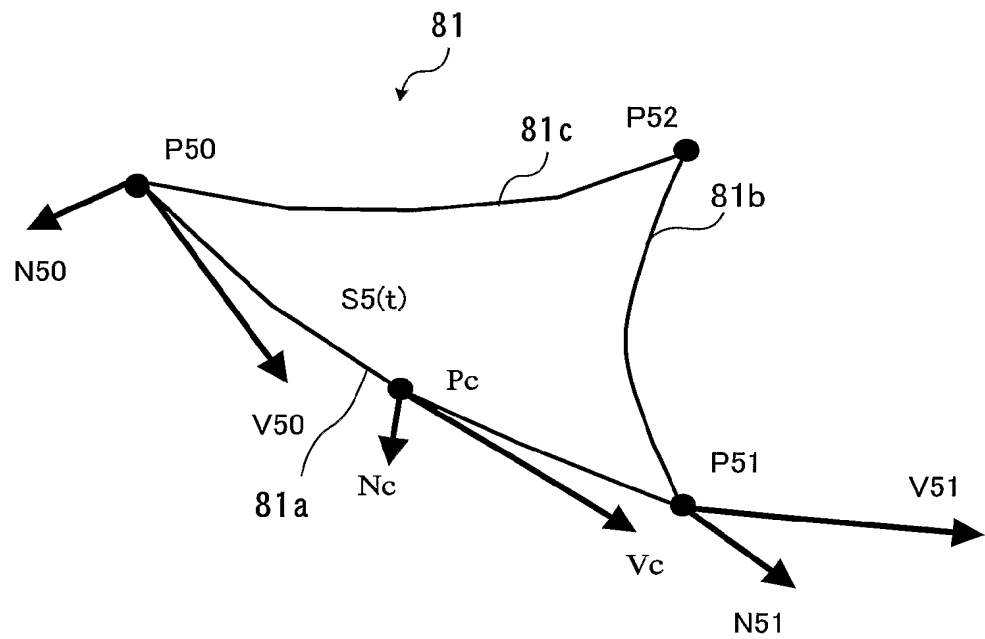


図22





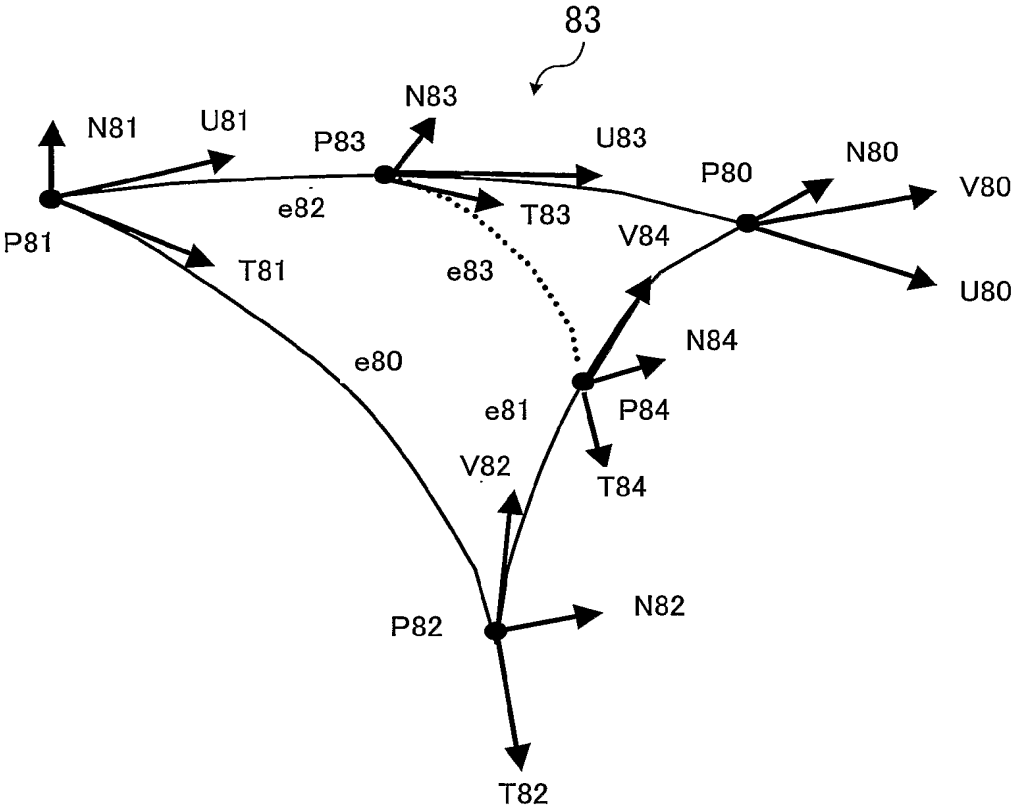


図24

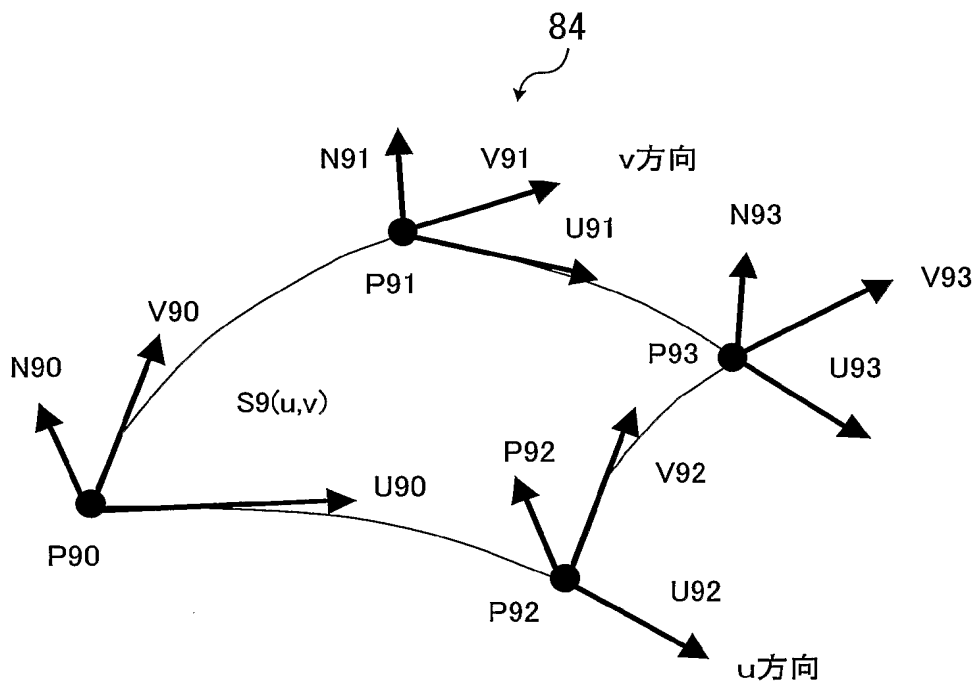
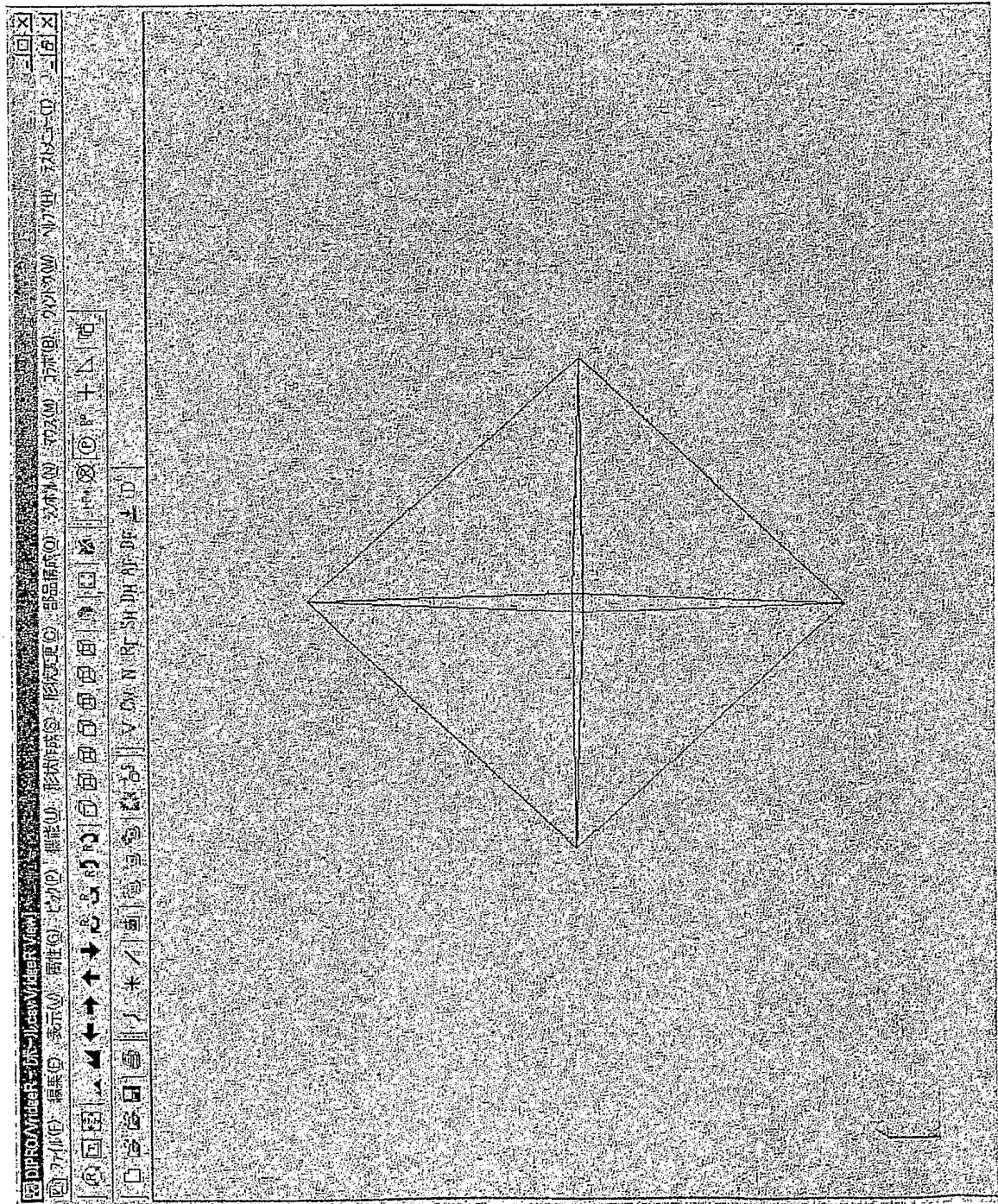
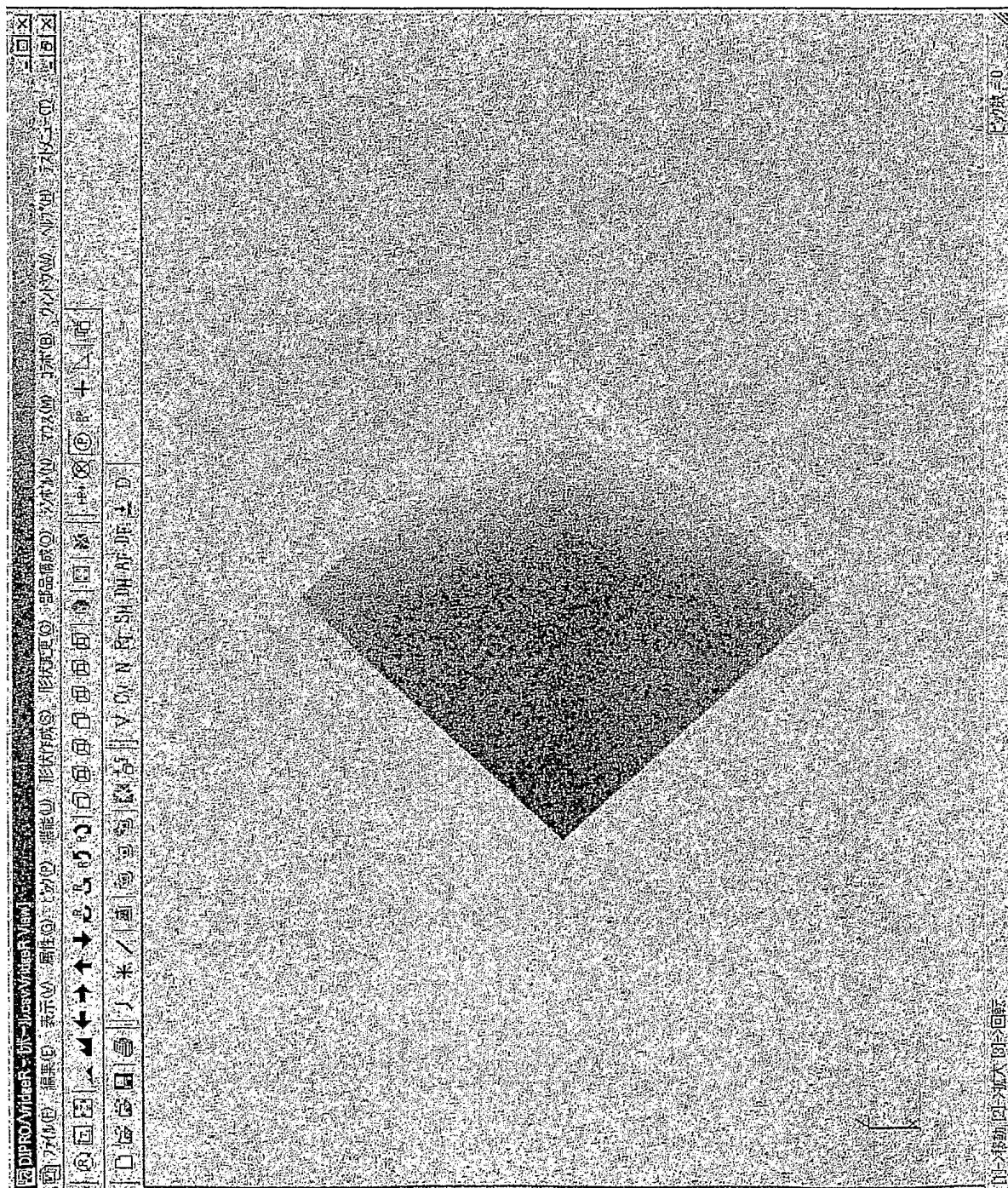
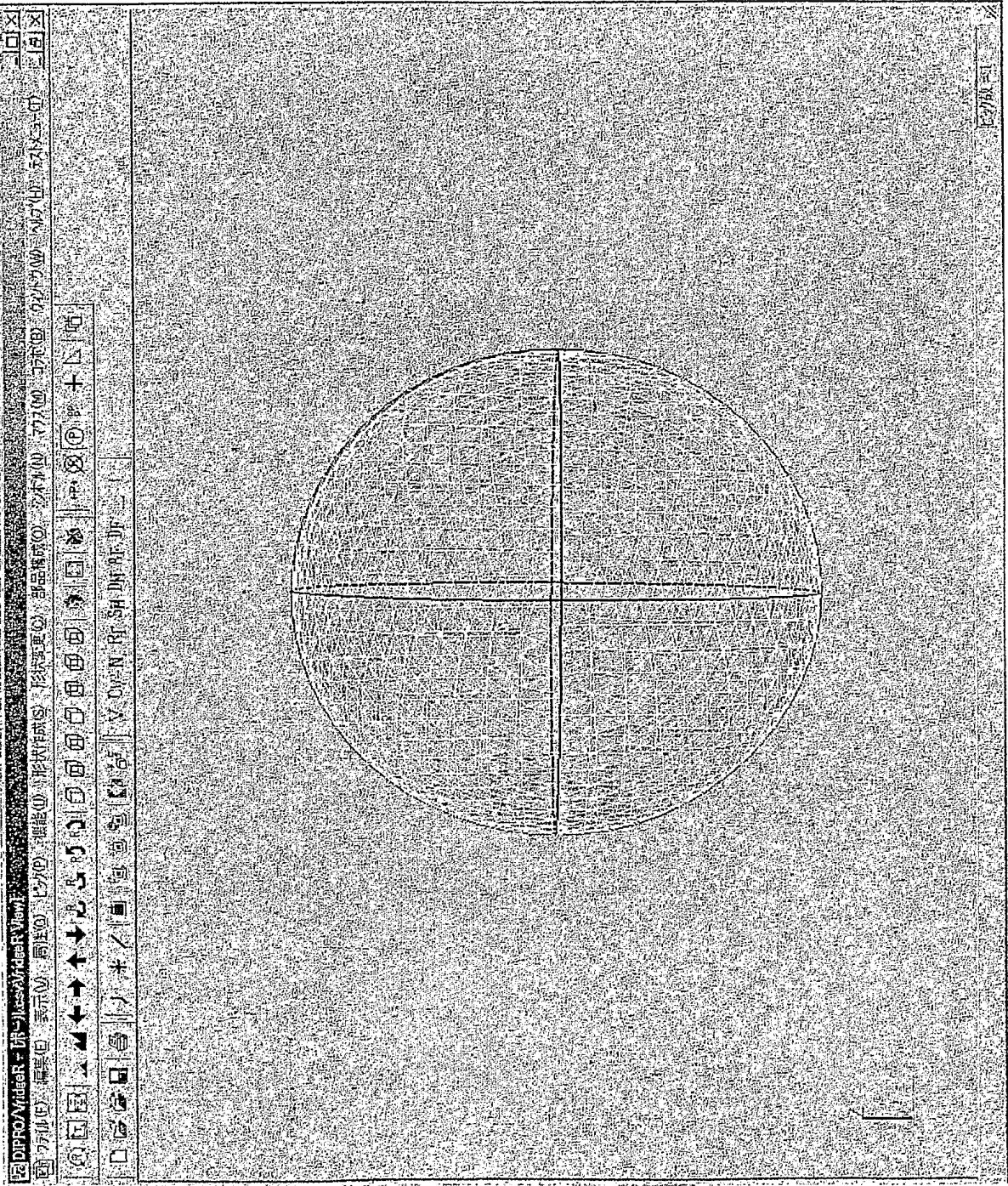


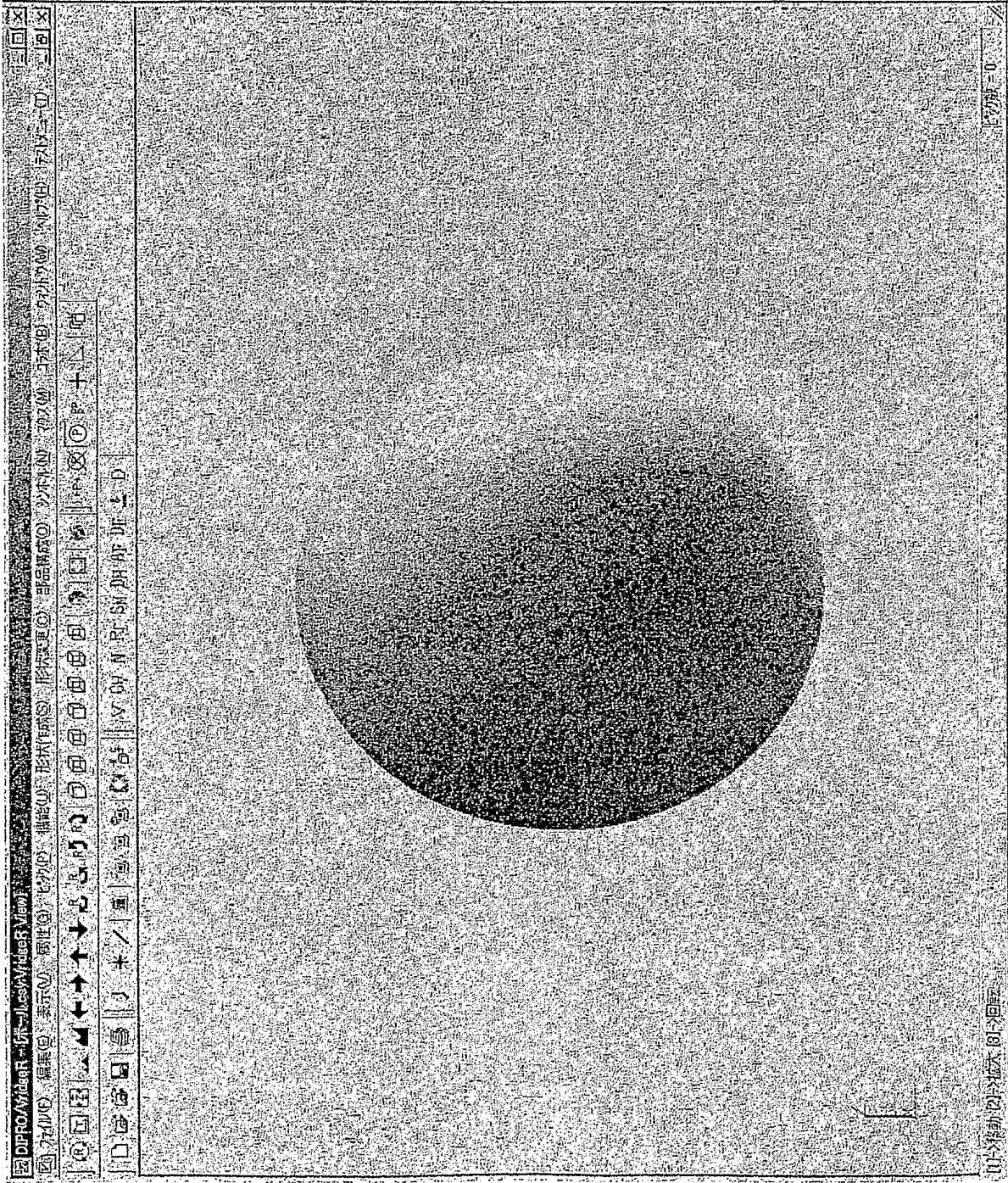
図25











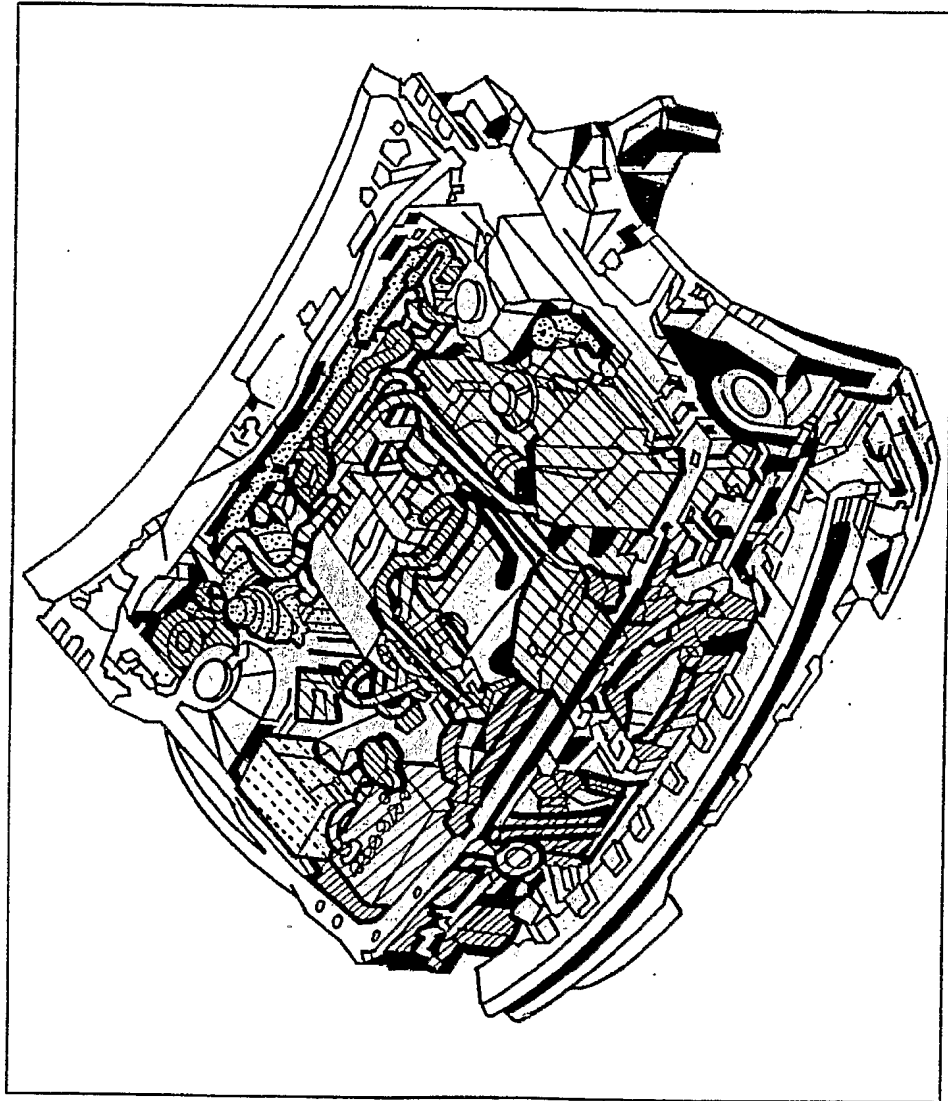


図30



# INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP02/01000

## A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl<sup>7</sup> G06T17/00, G06T17/40, G06F17/50

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl<sup>7</sup> G06T1/00, G06T11/00-17/50, G06F17/50

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Toroku Jitsuyo Shinan Koho	1994-2002
Kokai Jitsuyo Shinan Koho	1971-2002	Jitsuyo Shinan Toroku Koho	1996-2002

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JP 11-296686 A (Sony Computer Entertainment Inc.),	1-5, 7-10,
Y	29 October, 1999 (29.10.99), Full text; all drawings & WO 99/53445 A1 & EP 996094 A1 & US 2002/0005856 A1 & ZA 9902636 A & AU 9931670 A & BR 9906309 A & CN 1272933 A & MX 9911422 A1 & KR 2001013519 A & TW 430772 A	12-14 6, 11
Y	JP 2001-250129 A (Namco Ltd.), 14 September, 2001 (14.09.01), Page 7, right column, lines 35 to 38 (Family: none)	6, 11

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

\* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search  
26 April, 2002 (26.04.02)

Date of mailing of the international search report  
21 May, 2002 (21.05.02)

Name and mailing address of the ISA/  
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

## A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl<sup>7</sup> G06T17/00 G06T17/40 G06F17/50

## B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl<sup>7</sup> G06T1/00 G06T11/00-17/50 G06F17/50

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1922-1996年

日本国公開実用新案公報 1971-2002年

日本国登録実用新案公報 1994-2002年

日本国実用新案登録公報 1996-2002年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

## C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
X	JP 11-296686 A (株式会社ソニー・コンピュータエンタテインメント)	1-5,
	1999. 10. 29, 全文, 全図	7-10,
Y	& WO 99/53445 A1	12-14
	& EP 996094 A1	6, 11
	& US 2002/0005856 A1	
	& ZA 9902636 A	
	& AU 9931670 A	
	& BR 9906309 A	
	& CN 1272933 A	

☒ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

## \* 引用文献のカテゴリー

「A」 特に関連のある文献ではなく、一般的技術水準を示すもの

「E」 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)

「O」 口頭による開示、使用、展示等に言及する文献

「P」 国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&amp;」 同一パテントファミリー文献

国際調査を完了した日

26. 04. 02

国際調査報告の発送日

21.05.02

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

村松 貴士

5H

9854

電話番号 03-3581-1101 内線 3531

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
Y	& MX 9911422 A1 & KR 2001013519 A & TW 430772 A  JP 2001-250129 A (株式会社ナムコ) 2001. 09. 14 第7頁右欄第35行～第38行 (ファミリーなし)	6, 11